Leveraging Unpaired Data for the Creation of Controllable Digital Humans

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von
Soubhik Sanyal
aus Kolkata, Indien

Tübingen
2024

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	23.09.2024
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Hon.-Prof. Dr. Michael J. Black
2. Berichterstatter:	Prof. Dr. Andreas Geiger

To Tapati and Mahadeb, my parents

Abstract

Digital humans have grown increasingly popular, offering transformative potential across various fields such as education, entertainment, and healthcare. They enrich user experiences by providing immersive and personalized interactions. Enhancing these experiences involves making digital humans controllable, allowing for manipulation of aspects like pose and appearance, among others. Learning to create such controllable digital humans necessitates extensive data from diverse sources. This includes 2D human images alongside their corresponding 3D geometry and texture, 2D images showcasing similar appearances across a wide range of body poses, etc., for effective control over pose and appearance. However, the availability of such "paired data" is limited, making its collection both time-consuming and expensive. Despite these challenges, there is an abundance of unpaired 2D images with accessible, inexpensive labels—such as identity, type of clothing, appearance of clothing, etc. This thesis capitalizes on these affordable labels, employing informed observations from "unpaired data" to facilitate the learning of controllable digital humans through reconstruction, transposition, and generation processes.

The presented methods—RingNet, SPICE, and SCULPT—each tackles different aspects of controllable digital human modeling. RingNet (Sanyal et al. [2019]) exploits the consistent facial geometry across different images of the same individual to estimate 3D face shapes and poses without 2D-to-3D supervision. This method illustrates how leveraging the inherent properties of unpaired images—such as identity consistency—can circumvent the need for expensive paired datasets. Similarly, SPICE (Sanyal et al. [2021]) employs a self-supervised learning framework that harnesses unpaired images to generate realistic transpositions of human poses by understanding the underlying 3D body structure and maintaining consistency in body shape and appearance features across different poses. Finally, SCULPT (Sanyal et al. [2024] generates clothed and textured 3D meshes by integrating insights from unpaired 2D images and medium-sized 3D scans. This process employs an unpaired learning approach, conditioning texture and geometry generation on attributes easily derived from data, like the type and appearance of clothing.

In conclusion, this thesis highlights how unpaired data and innovative learning techniques can address the challenges of data scarcity and high costs in developing controllable digital humans by advancing reconstruction, transposition, and generation techniques.

Kurzfassung

Digitale Menschen erfreuen sich zunehmender Beliebtheit und bieten ein transformatives Potenzial in verschiedenen Bereichen wie Bildung, Unterhaltung und Gesundheitswesen. Sie bereichern das Nutzererlebnis, indem sie immersive und personalisierte Interaktionen ermöglichen. Um diese Erfahrungen zu verbessern, müssen digitale Menschen steuerbar gemacht werden, damit Aspekte wie Pose und Aussehen manipuliert werden können. Um zu lernen, wie man solche steuerbaren digitalen Menschen erschafft, sind umfangreiche Daten aus verschiedenen Quellen erforderlich. Dazu gehören 2D-Bilder von Menschen zusammen mit ihrer entsprechenden 3D-Geometrie und Textur, 2D-Bilder, die ein ähnliches Erscheinungsbild in einer Vielzahl von Körperhaltungen zeigen, usw., um eine effektive Kontrolle über Haltung und Aussehen zu ermöglichen. Die Verfügbarkeit solcher "gepaarten Daten" ist jedoch begrenzt, was ihre Erfassung sowohl zeitaufwändig als auch teuer macht. Trotz dieser Herausforderungen gibt es eine Fülle von ungepaarten 2D-Bildern mit zugänglichen, kostengünstigen Kennzeichnungen wie Identität, Art der Kleidung, Aussehen der Kleidung usw. Die vorliegende Arbeit nutzt diese kostengünstigen Beschriftungen, indem sie fundierte Beobachtungen aus "unpaired data" einsetzt, um das Lernen von kontrollierbaren digitalen Menschen durch Rekonstruktions-, Transpositions- und Generierungsprozesse zu erleichtern.

Die vorgestellten Methoden - RingNet, SPICE und SCULPT - befassen sich jeweils mit unterschiedlichen Aspekten der kontrollierbaren digitalen Menschmodellierung. RingNet (Sanyal et al. [2019]) nutzt die konsistente Gesichtsgeometrie in verschiedenen Bildern derselben Person, um 3D-Gesichtsformen und Posen ohne 2D-zu-3D-Überwachung zu schätzen. Diese Methode veranschaulicht, wie durch die Nutzung der inhärenten Eigenschaften von ungepaarten Bildern - wie z. B. der Identitätskonsistenz - die Notwendigkeit von teuren gepaarten Datensätzen umgangen werden kann. In ähnlicher Weise verwendet SPICE (Sanyal et al. [2021]) ein selbstüberwachtes Lernverfahren, das ungepaarte Bilder nutzt, um realistische Umsetzungen menschlicher Posen zu erzeugen, indem es die zugrunde liegende 3D-Körperstruktur versteht und die Konsistenz der Körperform und der Erscheinungsmerkmale über verschiedene Posen hinweg aufrechterhält. Schließlich generiert SCULPT (Sanyal et al. [2024] gekleidete und texturierte 3D-Netze durch die Integration von Erkenntnissen aus ungepaarten 2D-Bildern und mittelgroßen 3D-Scans. Dieser Prozess verwendet einen Ansatz des ungepaarten Lernens, bei dem die Textur- und Geometriegenerierung auf Attributen basiert, die sich leicht aus

den Daten ableiten lassen, wie z. B. die Art und das Aussehen der Kleidung.

Abschließend zeigt diese Arbeit auf, wie ungepaarte Daten und innovative Lerntechniken die Herausforderungen der Datenknappheit und der hohen Kosten bei der Entwicklung kontrollierbarer digitaler Menschen durch die Weiterentwicklung von Rekonstruktions-, Transpositions- und Generierungstechniken bewältigen können.

Acknowledgments

Alone we can do so little; together we can do so much.

– **Helen Keller**

I feel fortunate to have been supported by a wonderful group of people throughout my PhD journey. Without their help, this experience would not have been as enjoyable or informative.

First of all, I would like to express my gratitude to my advisor at MPI, Michael J. Black, for giving me the opportunity to join the vibrant Perceiving Systems group. Michael has been an incredibly supportive advisor on both a professional and personal level. His patience in listening to my ambitious research ideas, allowing me the freedom to choose my own research directions, and his insightful perspectives on the problems we faced, have helped me grow from a mere college student to a confident problem-solver. The best aspect of working with Michael is that he allowed me to fail while pursuing an idea and supported me in standing back up by teaching me how to learn from my failures. This support has been crucial in helping me get back on my feet and use what I've learned for future success. He has also been very supportive during my personal ups and downs and will always be a professional role model for me.

I am also deeply thankful to Timo Bolkart, my mentor, friend, and second advisor from the very beginning. Beneath his tough exterior, he has a big heart. Timo has been a steadfast supporter, collaborator, and the backbone of my research throughout my PhD. I have learned a lot from him, not just technically, but also about research ethics and conducting fair research. He is very committed to maintaining strong ethical standards in research. I consider him not only a mentor but also one of my best friends. One of the most valuable lessons he taught me is that one must be the strongest critic of their own work; if you are not convinced about your work, you can not convince others.

Throughout my PhD, I had the privilege of collaborating with brilliant researchers like Javier Romero and Justus Thies. Their guidance and technical expertise have greatly enriched my experience, helping me lead various projects and achieve my goals. I am deeply grateful for their exceptional support. I also want to thank Leila and Sara from the IMPRS-IS school for their support and for keeping me on track for graduation. Additionally, I am thankful to Andreas and Matthias for serving on my TAC committee and providing valuable feedback

on my research and career. My thesis would not have been complete without the excellent support from Tsvetelina, Taylor, Galina, Markus, and Claudia of the PS Data Capture team, as well as Benjamin for IT support. Special thanks to Tsvetelina for being there from day one.

My journey would have been much more challenging without the support of my wonderful colleagues and friends, both within and outside the PS Department. I am deeply grateful to Partha, Marylin, Lea, Caroline, Nikos, Markos, Mohammed, Dimitris, Daniel, Anurag, Yao, Yuliang, Jinlong, Paola, Omid, Sai, Artur, Rama, Osman, Quanli, Vassilis, Nadine, Shashank, Dafni, Omri, Radek, Valeria, Caterina, Sarah, Peter, Gerogio, Thasos, Joachim, Prianka, Camila, Haiwen, Dominik, Laila, Kyle, Vini, and Bala (listed in no particular order, and I hope I have not missed anyone). Over the years, Partha, Marylin, Lea, and Caroline have become very close to me, and I look forward to our friendship continuing into the next phase of our lives. This section would be incomplete without Simon and Celina, who became close friends during this journey. I wish them all the best in their new journey with their baby, Hanes. I consider Nicole and Melanie more as friends than just PS admins. Their countless emotional, administrative, and friendly support was crucial to my positive experience in Germany. A special thanks to them.

I am grateful to my cousin sisters Moudidi and Dolu for the family gossip from India, which always brings a smile to my face. I also thank Rinididi and Anupda for their great support during family needs while I was abroad. I am deeply thankful to my uncle, Mrinmoy Bhattacharya, who has been an inspiration since my childhood. His ideologies, thoughtfulness, and satirical sense of humor have profoundly influenced me and helped shape part of my character.

Finally, I must express my deep gratitude to my mother and father. They are the pillars of support in my life, and without them, I would not be where I am today. Words cannot adequately describe their profound influence on my life, personality, and moral values, all of which are reflected in my PhD journey and my scientific integrity.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, the concept of digital humans has emerged as a groundbreaking advancement, with its influence pervading numerous sectors including education, entertainment, and healthcare. Digital humans, characterized by their virtual representation of humans in digital form, offer a myriad of possibilities for enriching user experiences through immersive and personalized interactions. Their appeal lies in their ability to simulate real human behaviors, emotions, and interactions, thereby bridging the gap between the virtual and the real world. In educational settings, they can serve as virtual tutors, providing personalized learning experiences. In entertainment, they bring characters to life in a more relatable and interactive manner. In healthcare, they can be utilized for therapeutic purposes, patient education, and as virtual caregivers. The potential applications are as vast as they are transformative, indicating a significant shift in how digital content is consumed and interacted with.

Enhancing user experiences with digital humans significantly hinges on the ability to render these entities controllable. Controllability, in this context, refers to the manipulation of various attributes of a digital human's appearance, including pose, facial expressions, and clothing, among others. The capacity to adjust facial expressions, for instance, allows for the transfer of facial shapes and expressions from images and videos onto 3D models, which are subsequently utilized in graph-

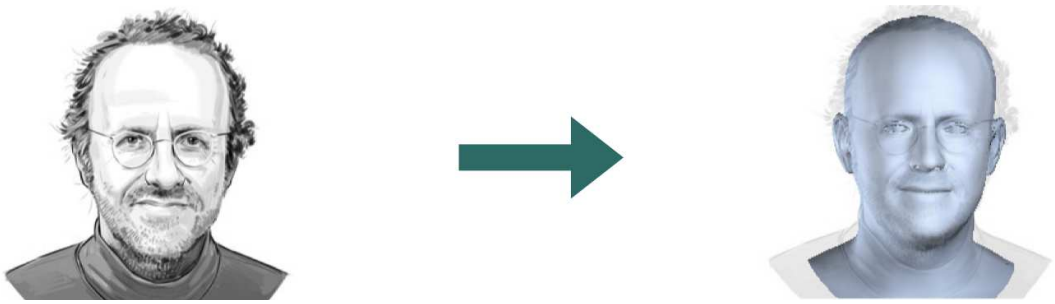


Figure 1.1: **Reconstruction:** Given an image of a human face (left) the goal is to reconstruct the 3D facial shape, pose, and expression (right) of that person.

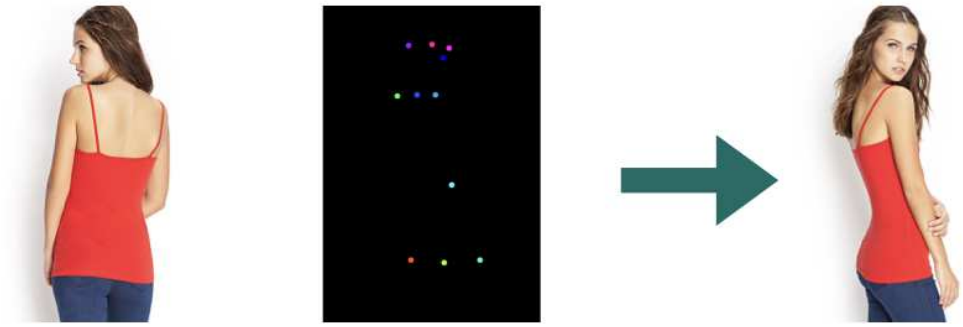


Figure 1.2: **Transposition:** Given an image of a person in a source pose (left) and a target pose (middle), the goal is to synthesize the image of the person in the target pose (right) while keeping the appearance unchanged.

ics and animation sectors. Such a process requires a method (Sanyal et al. [2019]) capable of reconstructing the facial shape, pose, and expression from static images and videos (Fig. 1.1).

Further, the capability to alter poses while maintaining the appearance of the digital human paves the way for e-commerce sectors to generate short, moving videos from static images (Fig. 1.2). This innovation holds the potential to enhance apparel sales, necessitating a transposition algorithm (Sanyal et al. [2021]) that can synthesize novel human poses while preserving the same appearance. Additionally, controlling the distribution of clothing in terms of geometry, texture, and overall appearance (Fig. 1.3) enables rapid changes in outfits, facilitating the creation of virtual avatars for uses ranging from gaming engines to movies and e-commerce platforms. This kind of customization requires a generative model adept at learning the diverse spaces of geometry and appearance associated with controllable clothed human modeling (Sanyal et al. [2024].

Creating this level of control needs large and varied datasets. This data includes 2D pictures of people, their 3D shapes and textures, and pictures showing different looks for many body poses. For instance, to make controllable faces, we need a dataset with 2D pictures and their matching 3D scans, like the left-most pair in Fig. 1.4, to train the model. Also, to teach a model to change poses but keep the appearance the same, we need pairs of data showing the same person in different poses but looking similar (the middle pair in Fig. 1.4 is an example). We need many examples of people in different poses but with similar looks.

Similarly, to train a model on changing clothes in 3D, we need lots of data on the shapes (meshes) of clothed humans and their appearance (textures), like the right-most pair of images in Fig. 1.4 shows. These big datasets are crucial for training models to accurately handle how people look and their poses, giving us

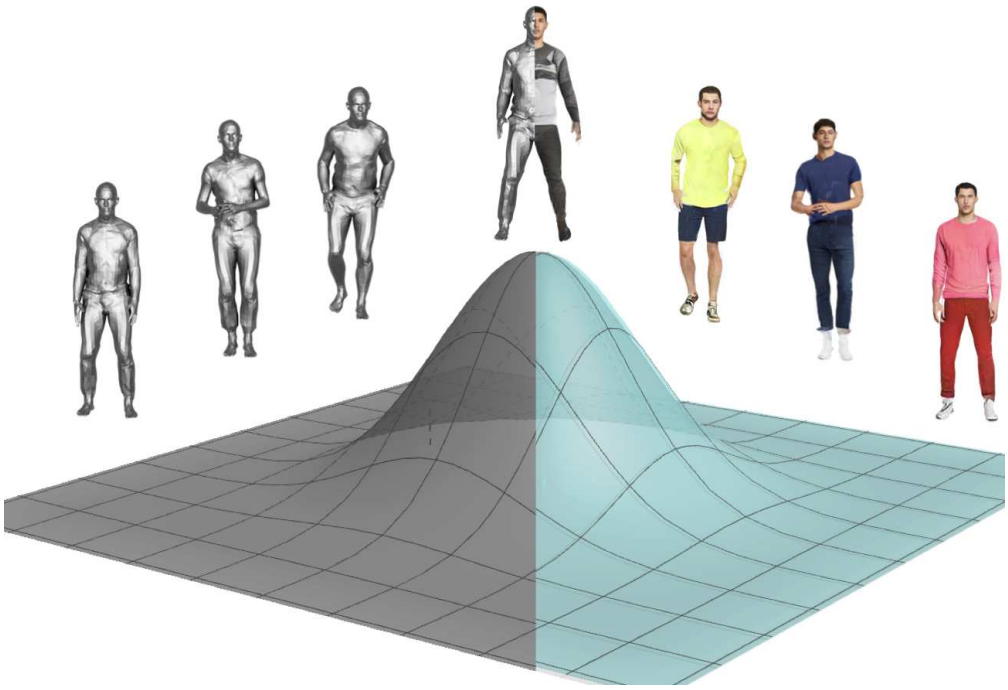


Figure 1.3: **Generation:** The goal is to learn the geometry and appearance distribution of the clothed digital avatars.

precise control over their digital versions. But getting this "paired data" is tough. Not only is it scarce, but the process of collecting it is both time-consuming and costly, posing barriers to research and development in this field.

Despite facing challenges, the abundance of readily available 2D images offers a unique opportunity. These images can be inexpensively labeled with various types of information, such as identity, clothing type, and appearance. Leveraging these labels allows us to make informed observations from unpaired data, which helps in the learning process for creating controllable digital humans. For instance, our reconstruction approach, RingNet (Sanyal et al. [2019]), utilizes the identity information of individuals to estimate their 3D face shapes, poses, and expressions without needing direct 2D-to-3D supervision. The underlying principle is that having multiple images of the same person provides a strong constraint on the 3D shape of their face, as the shape remains constant while other factors like pose, lighting, and expression may change. RingNet processes multiple images of an individual, ensuring the shape consistency across all pairs of images while minimizing the error between observed and projected 3D landmarks.

Similarly, our pose transposition method, SPICE (Sanyal et al. [2021]), adopts a self-supervised learning framework to bypass the necessity for paired images. It leverages the inherent 3D body structure to maintain consistency in body shape

Figure 1.4: **"Paired data":** From left to right, each pair of images represents a specific set. The left-most pair consists of a 2D facial image and its corresponding 3D mesh. The middle pair of images represents a set showing the source and target poses of a single individual, maintaining the same appearance. The right-most pair represents a set of registered meshes along with their corresponding textures.

and appearance features across different poses during model training. This approach includes incorporating 3D information through a pose loss that aligns the body pose in the generated image with the target pose in 3D, and a shape consistency loss that ensures the person in the generated image maintains the same 3D shape as in the source image. Additionally, we introduce a pose-dependent appearance consistency to ensure that the body surface's projected appearance in two different poses matches for the corresponding body parts. SPICE has shown superior performance compared to other unsupervised methods and rivals that of the leading supervised methods at the time of publication.

Continuing this line, SCULPT (Sanyal et al. [2024]) generates clothed and textured 3D meshes by harnessing insights from unpaired 2D images and medium-sized 3D scans. This process utilizes an unpaired learning approach, basing the generation of texture and geometry on easily extractable attributes, such as clothing type and appearance. The training of the geometry model utilizes the CAPE dataset (Ma et al. [2020b]), which includes pose annotations and registered SMPL meshes (Loper et al. [2015]). To ensure the consistency of appearance and geometry, the texture generator, trained on a vast collection of 2D fashion images, is conditioned on the pre-trained and fixed geometry model. By conditioning both texture and geometry generators with attribute labels, we minimize the entanglement between pose, clothing type, and appearance, leading to improved performance in geometry and competitive appearance results compared to existing methods.

Overall, this thesis explores the potential of utilizing unpaired data for the reconstruction, transposition, and generation of digital humans, aiming to bypass the limitations imposed by the scarcity of paired data. Through these strategies, we seek to push the boundaries of controlling digital human creation.

Outlook: The subsequent chapters are organized to systematically address the three key aspects of digital human creation: reconstruction, transposition, and generation. Chapter 2 delves into the intricacies of reconstructing a full 3D face, complete with head and neck, from single-view images. Chapter 3 explores the process of synthesizing human images in unique poses and specific appearances. Chapter 4 concentrates on generating 3D textured human mesh models, a critical aspect of creating realistic digital humans.

Each chapter is divided into five sections. We commence with a brief introduction and rationale for the topic explored in the respective chapter, setting the stage for the subsequent discussion. This is followed by a comprehensive literature review on the relevant subject matter, offering insights into the current state of research and identifying gaps in knowledge. Next, we provide an in-depth analysis of the technical and scientific aspects of the proposed method, detailing its inner workings and potential advantages. Our approach is then assessed through extensive experimentation, using a variety of datasets and performance metrics to determine its efficacy. We conclude each chapter with a concise summary of the content covered, highlighting the key findings and implications for the field.

Chapter 5 revisits the contributions made throughout the thesis, synthesizing the insights from previous chapters and underscoring their collective significance. Furthermore, this chapter discusses potential future research directions, identifying areas where additional exploration could lead to further advancements in digital human creation.

Chapter 2

Reconstruction leveraging unpaired data

2.1 Introduction

Here the goal is to estimate 3D head and face shape from a single image of a person. In contrast to previous methods, we are interested in more than just a tightly cropped region around the face. Instead, we estimate the full 3D face, head and neck. Such a representation is necessary for applications in VR/AR, virtual glasses try-on, animation, biometrics, etc. Furthermore, we seek a representation that captures the 3D facial expression, factors face shape from expression, and can be reposed and animated. While there have been numerous methods proposed in the computer vision literature to address the problem of facial shape estimation (Zollhöfer et al. [2018]), no previous methods address all of our goals.

Specifically, we train a neural network that regresses from image pixels directly to the parameters of a 3D face model. Here we use FLAME (Li et al. [2017a]) because it is more accurate than other models, captures a wide range of shapes, models the whole head and neck, can be easily animated, and is freely available. Training a network to solve this problem, however, is challenging because there is little paired data of 3D heads/faces together with natural images of people. For robustness to imaging conditions, pose, facial hair, camera noise, lighting, etc., we wish to train from a large corpus of in-the-wild images. Such images, by definition, lack controlled ground truth 3D data.

This is a generic problem in computer vision – finding 2D training data is easy but learning to regress 3D from 2D is hard when paired 3D training data is very limited and difficult to acquire. Without ground truth 3D, there are several options but each has problems. Synthetic training data typically does not capture real-world complexity. One can fit a 3D model to 2D image features but this mapping is ambiguous and, consequently, inaccurate. Because of the ambiguity, training a neural network using only a loss between observed 2D, and projected 3D, features does not lead to good results (Kanazawa et al. [2018]).

Figure 2.1: **RingNet** learns a mapping from the pixels of a single image to the 3D facial parameters of the FLAME model (Li et al. [2017a]) without 3D supervision. First and third row: Images are from the CelebA dataset (Liu et al. [2015]). Second and foruth row: estimated shape, pose and expression. Additional results can be found in Appendix A.

To address the lack of training data, we propose a new method that learns the mapping from pixels to 3D shape *without any supervised 2D-to-3D training data.* To do so, we learn the mapping using only 2D facial features, automatically extracted with OpenPose (Simon et al. [2017]). To make this possible, our key observation is that multiple images of the same person provide strong constraints on 3D face shape because the shape remains constant although other things may change such as pose, lighting, and expression. FLAME factors pose and shape, allowing our model to learn what is constant (shape) and factor out what changes (pose and expression).

While it is a fact that face shape is constant for an individual across images, we need to define a training approach that lets a neural network exploit this shape constancy. To that end, we introduce *RingNet*. RingNet takes multiple images of a person and enforces that the shape should be similar between all pairs of images, while minimizing the 2D error between observed features and projected 3D features. While this encourages the network to encode the shapes similarly, we find this is not sufficient. We also add to the "ring" a face belonging to a different random person and enforce that the distance in the latent space between all other images in the ring is larger than the distance between the same person. Similar ideas have been used in manifold learning (e.g. triplet loss) (Weinberger and Saul [2006]) and face recognition (Schroff et al. [2015]), but, to our knowledge, our approach has not previously been used to learn a mapping from 2D to 3D geometry. We find that going beyond a triplet to a larger ring, is critical in learning accurate geometry.

While we train with multiple images of a person, note that, at run time, we only need a single image. With this formulation, we are able to train a network to regress the parameters of FLAME directly from image pixels. Because we train this with "in the wild" images, the network is robust across a wide range of conditions as illustrated in Fig. 2.1. The approach is more general, however, and could be applied to other 2D-to-3D learning problems.

Evaluating the accuracy of 3D face estimation methods remains a challenge and, despite many methods that have been published, there are no rigorous comparisons of 3D accuracy across a wide range of imaging conditions, poses, lighting and occlusion. To address this, we collected a new dataset called *NoW (Not quite in-the-Wild)*, with high-resolution ground truth scans and high-quality images of 100 subjects taken in a range of conditions (Fig. 2.2). NoW is more complex than previous datasets and we use it to evaluate all recent methods with publicly available implementations. Specifically we compare with Tuan Tran et al. [2017], Trân et al. [2018] and Feng et al. [2018a], which are trained with 3D supervision. Despite not having any 2D-to-3D supervision our RingNet method recovers more accurate 3D face shape. We also evaluate the method qualitatively on challenging

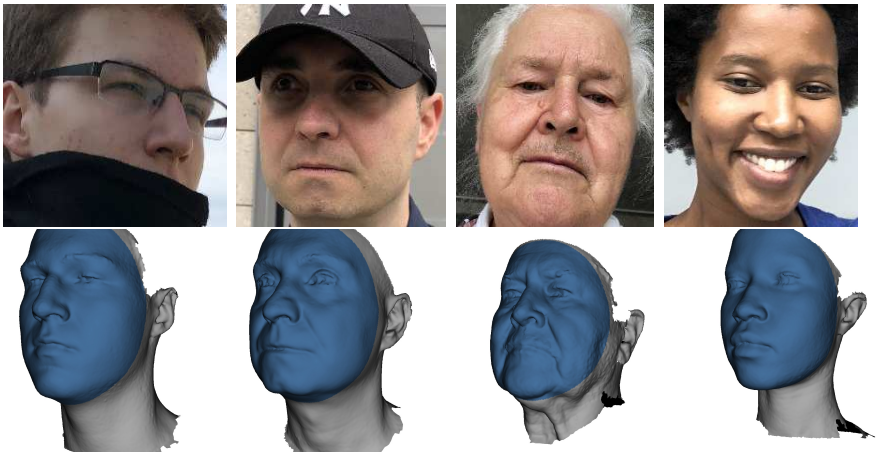


Figure 2.2: **NoW dataset**: The **NoW dataset** includes a variety of images taken in different conditions (top) and high-resolution 3D head scans (bottom). The dark blue region is the part we considered for the face challenge.

in-the-wild face images.

In summary, here the main contributions are: (1) Full face, head with neck reconstruction from a single face image. (2) RingNet – an end-to-end trainable network that enforces shape consistency across face images of the subject with varying viewing angle, light conditions, resolution and occlusion. (3) A novel shape consistency loss for learning 3D geometry from 2D input. (4) NoW – a benchmark dataset for qualitative and quantitative evaluation of 3D face reconstruction methods. (5) Finally, we make the model, training code, and new dataset freely available for research purposes to encourage quantitative comparison.

2.2 Related Work

There are several approaches to the problem of 3D face estimation from images. One approach estimates depth maps, normals, etc.; that is, these methods produce a representation of object shape tied to pixels but specialized for faces. The other approach estimates a 3D shape model that can be animated. We focus on methods in the latter category. In a recent review paper, Zollhöfer et al. [2018] describe the state of the art in monocular face reconstruction and provide a forward-looking set of challenges for the field. Note, that the boundary between supervised, weakly supervised, and unsupervised methods is a blurry one. Most methods use some form of 3D shape model, which is learned from scans in advance; we do not call this supervision here. Here the term supervised implies that paired 2D-to-3D data is used; this might be from real data or synthetic data. If a 3D model is first

optimized to fit 2D image features, then we say this uses 2D-to-3D supervision. If 2D image features are used but there is no 3D data in training the network, then this is weakly supervised in general and unsupervised relative to the 2D-to-3D task.

Face recognition: Given our model's reliance on facial identity information for training, it is imperative to reference recent works in face recognition that can extract identity information from images, ensuring the comprehensiveness of this chapter. This domain has undergone extensive research for an extended period, witnessing significant advancements from the pre-deep learning era (Biswas et al. [2011], Sanyal et al. [2015], Mudunuri and Biswas [2015], Sanyal et al. [2017a], Sanyal et al. [2017b], etc.) to the current deep learning era (Deng et al. [2019a], Wang and Deng [2021], Kim et al. [2022], etc.). Nowadays, face recognition systems are capable of predicting identity with high accuracy. For further details, readers are encouraged to consult the cited papers.

Quantitative evaluation: Quantitative comparison between methods has been limited by a lack of common datasets with complex images and high-quality ground truth 3D scans. Recently, Feng et al. [2018b] organized a single-image to 3D-face reconstruction challenge where they provided the ground truth scans for subjects. Our NoW benchmark is complementary to this method as its focus is on extreme viewing angles, facial expressions, and partial occlusions.

3DMM Head models: Most current shape models are descendants of the original Blanz and Vetter 3D morphable model (3DMM) (Blanz and Vetter [1999]). While there are many variations and improvements to this model such as Gerig et al. [2018], we use FLAME (Li et al. [2017a]) here because both the shape space and expression space are trained from more scans than other methods. Only FLAME includes the neck region in the shape space and models the pose-dependent deformations of the neck with head rotation.

Optimization with tightly cropped faces: Most existing methods require tightly cropped input images and/or reconstruct only a tightly cropped region of the face for which existing shape priors are appropriate. Tightly cropped face regions make the estimation of head rotation ambiguous. Until very recently, this has been the dominant paradigm (Bas et al. [2016], Suwajanakorn et al. [2014], Garrido et al. [2016]). For example, Kemelmacher-Shlizerman and Seitz [2011] use multi-image shading to reconstruct from collection of images allowing changes in viewpoint and shape. Thies et al. [2016] achieve accurate results on monocular

video sequences. While these approaches can achieve good results with high-realism, they are computationally expensive.

Learning with 3D supervision: Deep learning methods are quickly replacing the optimization-based approaches (Trân et al. [2018], Zhu et al. [2016], Kim et al. [2018], Jackson et al. [2017]). For example, Sela et al. [2017] use a synthetic dataset to generate an image-to-depth mapping and a pixel-to-vertex mapping, which are combined to generate the face mesh. Tuan Tran et al. [2017] directly regress the 3DMM parameters of a face model with a dense network. Their key idea is to use multiple images of the same subject and fit a 3DMM to each image using 2D landmarks. They then take a weighted average of the fitted meshes to use it as the ground truth to train their network. Feng et al. [2018a] regress from image to a UV position map that records the position information of the 3D face. All the aforementioned methods use some form of 3D supervision like synthetic rendering, optimization-based fitting of a 3DMM, or a 3DMM to generate UV maps or volumetric representation. None of the fitting-based methods produce true ground truth for real world face images, while synthetically generated faces may not generalize well to the real world (Tewari et al. [2018]). Methods that rely on fitting a 3DMM to images using 2D-3D correspondences to create a pseudo ground truth are always limited by the expressiveness of the 3DMM and the accuracy of the fitting process.

Learning with synthetic data supervision: Sengupta et al. [2018] learn to mimic a Lambertian rendering process by using a mixture of synthetically rendered images and real images. They work with tightly cropped faces and do not produce a model that can be animated. Genova et al. [2018] propose an end-to-end learning approach using a differentiable rendering process. They also train their encoder using synthetic data and its corresponding 3D parameters. Tran and Liu [2018] learn a nonlinear 3DMM model by using an analytically differentiable rendering layer and in a weakly supervised fashion with 3D data. At the time this work was published, the synthetic data utilized in various studies were of low quality, leading to inadequate supervision with synthetic data. However, recent advancements (Wood et al. [2021], Wood et al. [2022]) have demonstrated that high-quality synthetic data can effectively contribute to solving the task at hand.

Learning with no 3D supervision: MoFA (Tewari et al. [2017]) estimates the parameters of a 3DMM and is trained end-to-end using a photometric loss and an optional 2D feature loss. It is effectively a neural network version of the original Blanz and Vetter model in that it models shape, skin reflectance, and illumination to produce a realistic image that is matched to the input. The advantage of this

is that the approach is significantly faster than optimization methods (Tewari et al. [2018]). MoFA estimates a tight crop of the face and produces good looking results but has trouble with extreme expressions. They only perform quantitative evaluation on real images using the FaceWarehouse model as the "ground truth"; this is not an accurate representation of true 3D face shape.

The methods that learn without any 2D-to-3D supervision all explicitly model the image formation process (like Blanz and Vetter) and formulate a photometric loss and typically also incorporate 2D face feature detections with known correspondence to the 3D model. The problem with the photometric loss is that the model of image formation is always approximate (e.g. Lambertian). Ideally, one would like a network to learn not just about face shape but about the complexity of real world images and how they relate to shape. To that end, our RingNet approach uses only the 2D face features and no photometric term. Despite (or because of) this, the method is able to learn a mapping from pixels directly to 3D face shape. This was the least supervised of published methods, at the time of publication.

2.3 Method

The goal of our method is to estimate 3D head and face shape from a single face image I. Given an image, we assume the face is detected, loosely cropped, and approximately centered. During training, our method leverages 2D landmarks and identity labels as input. During inference it uses only image pixels; 2D landmarks and identity labels are not used.

Key idea: The key idea can be summarized as follows: 1) The face shape of a person remains unchanged, even though an image of the face may vary in viewing angle, lighting condition, resolution, occlusion, expression, or other factors. 2) Every person has a unique face shape (not considering identical twins).

We leverage this idea by introducing a shape consistency loss, embodied in our ring-structured network. RingNet (Fig. 2.3) is a multiple encoder-decoder based architecture, with weight sharing between the encoders, and shape constraints on the shape variables. Each encoder in the ring is a combination of a feature extractor network and a regressor network. Imposing shape constraints on the shape variables forces the network to disentangle facial shape, expression, head pose, and camera parameters. We use FLAME (Li et al. [2017a]) as a decoder to reconstruct 3D faces from the semantically meaningful embedding, and to obtain a decoupling within the embedding space into semantically meaningful parameters (i.e. shape, expression, and pose parameters).

We introduce the FLAME decoder, the RingNet architecture, and the losses in more detail in the following.

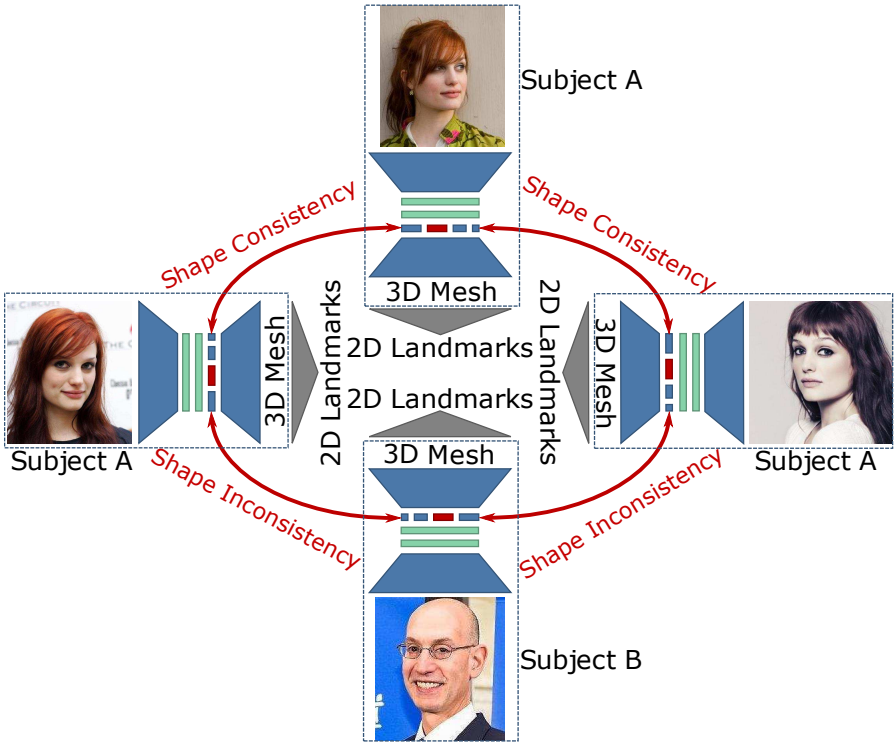


Figure 2.3: **Overview of RingNet:** RingNet takes multiple images of the same person (Subject A) and an image of a different person (Subject B) during training and enforces shape consistency between the same subjects and shape inconsistency between the different subjects. The computed 3D landmarks from the predicted 3D mesh are projected into the 2D domain to compute the loss with ground-truth 2D landmarks. During inference, RingNet takes a single image as input and predicts the corresponding 3D mesh. Images are taken from Cao et al. [2018]. The figure is a simplified version for illustration purposes.

2.3.1 FLAME model

FLAME uses linear transformations to describe identity and expression dependent shape variations, and standard linear blend skinning (LBS) to model neck, jaw, and eyeball rotations around $K = 4$ joints. Parametrized by coefficients for shape, $\boldsymbol{\beta} \in \mathbb{R}^{|\boldsymbol{\beta}|}$, pose $\boldsymbol{\theta} \in \mathbb{R}^{3K+3}$, and expression $\boldsymbol{\psi} \in \mathbb{R}^{|\vec{\psi}|}$, FLAME returns $N = 5023$ vertices. FLAME models identity-dependent shape variations $B_S(\boldsymbol{\beta}; \mathcal{S}) : \mathbb{R}^{|\boldsymbol{\beta}|} \rightarrow \mathbb{R}^{3N}$, corrective pose blendshapes $B_P(\boldsymbol{\theta}; \mathcal{P}) : \mathbb{R}^{3K+3} \rightarrow \mathbb{R}^{3N}$, and expression blendshapes $B_E(\boldsymbol{\psi}; \mathcal{E}) : \mathbb{R}^{|\vec{\psi}|} \rightarrow \mathbb{R}^{3N}$ as linear transformations with learned bases $\mathcal{S}$, $\mathcal{E}$, and $\mathcal{P}$. Given a template $\overline{\mathbf{T}} \in \mathbb{R}^{3N}$ in the "zero pose", identity, pose, and expression blendshapes, are modeled as vertex offsets from $\overline{\mathbf{T}}$.

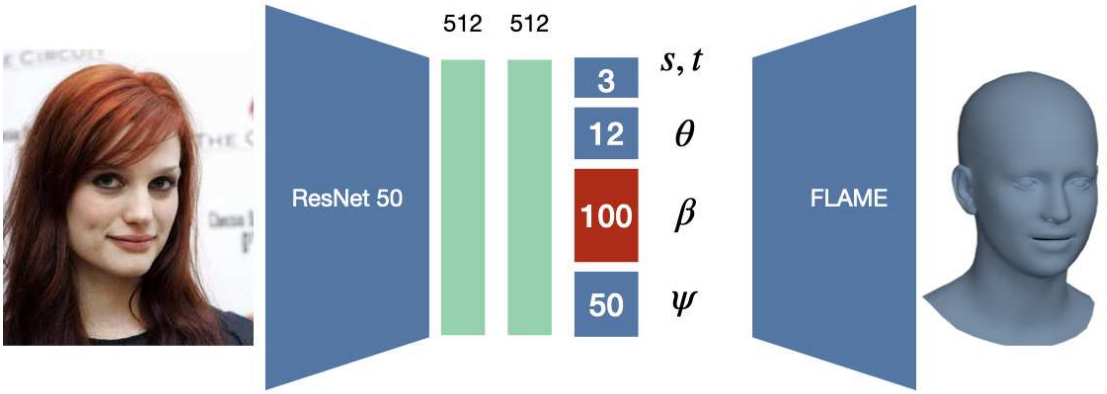


Figure 2.4: **Ring element:** A single ring element that outputs a 3D mesh for a monocular image.

Each of the pose vectors $\boldsymbol{\theta} \in \mathbb{R}^{3K+3}$ contains (K+1) rotation vectors in an axis-angle representation; i.e. one vector per joint plus the global rotation. The blend skinning function $W(\overline{\mathbf{T}}, \mathbf{J}, \boldsymbol{\theta}, \mathcal{W})$ then rotates the vertices around the joints $\mathbf{J} \in \mathbb{R}^{3K}$, linearly smoothed by blendweights $\mathcal{W} \in \mathbb{R}^{K \times N}$. More formally, FLAME is given as

$$M(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}) = W(T_P(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}), \mathbf{J}(\boldsymbol{\beta}), \boldsymbol{\theta}, \mathcal{W}), \tag{2.1}$$

with

$$T_P(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}) = \overline{\mathbf{T}} + B_S(\boldsymbol{\beta}; \mathcal{S}) + B_P(\boldsymbol{\theta}; \mathcal{P}) + B_E(\boldsymbol{\psi}; \mathcal{E}). \tag{2.2}$$

The joints are defined as a function of $\boldsymbol{\beta}$ since different face shapes require different joint locations. We use Equation 2.1 for decoding our embedding space to generate a 3D mesh of a complete head and face.

2.3.2 RingNet

The recent advances in face recognition (e.g. Zhang et al. [2017]) and facial landmark detection (e.g. Bulat and Tzimiropoulos [2017], Simon et al. [2017]) have led to large image datasets with identity labels and 2D face landmarks. For training, we assume a corpus of 2D face images I_i, corresponding identity labels c_i, and landmarks k_i.

The shape consistency assumption can be formalized by $\boldsymbol{\beta}_i = \boldsymbol{\beta}_j, \forall c_i = c_j$ (i.e. the face shape of one subject should remain the same across multiple images) and $\boldsymbol{\beta}_i \neq \boldsymbol{\beta}_j, \forall c_i \neq c_j$ (i.e. the face shape of different subjects should be distinct). RingNet introduces a ring-shaped architecture that jointly optimizes for shape consistency for an arbitrary number of input images in parallel. For details regarding the shape consistency, see Section 2.3.3.

RingNet is divided into R ring elements $e_{i=1}^{i=R}$ as shown in Figure 2.3, where each e_i consists of an encoder and a decoder network (see Figure 2.4). The encoders share weights across e_i, the decoder weights remain fixed during training. The encoder is a combination of a feature extractor network f_{feat} and regression network f_{reg}. Given an image I_i, f_{feat} outputs a high-dimensional vector, which is then encoded by f_{reg} into a semantically meaningful vector (i.e., $f_{\text{enc}}(I_i) = f_{\text{reg}}(f_{\text{feat}}(I_i))$). This vector can be expressed as a concatenation of the camera, pose, shape, and expression parameters, i.e., $f_{\text{enc}}(I_i) = [\text{cam}_i, \boldsymbol{\theta}_i, \boldsymbol{\beta}_i, \boldsymbol{\psi}_i]$, where $\boldsymbol{\theta}_i, \boldsymbol{\beta}_i, \boldsymbol{\psi}_i$ are FLAME parameters.

For simplicity we omit I in the following and use $f_{\text{enc}}(I_i) = f_{\text{enc},i}$ and $f_{\text{feat}}(I_i) = f_{\text{feat},i}$. The regression network iteratively regresses $f_{\text{enc},i}$ in an iterative error feedback loop (Kanazawa et al. [2018], Carreira et al. [2016]), instead of directly regressing $f_{\text{enc},i}$ from $f_{\text{feat},i}$. In each iteration step, progressive shifts from the previous estimate are made to reach the current estimate. Formally the regression network takes the concatenated $[f^t_{\text{feat},i}, f^t_{\text{enc},i}]$ as input and gives $\delta f^t_{\text{enc},i}$ as output. Then we update the current estimate by,

$$f^{t+1}_{\text{enc},i} = f^t_{\text{enc},i} + \delta f^t_{\text{enc},i}. \tag{2.3}$$

This iterative network performs multiple regression iterations per iteration of the entire RingNet training. The initial estimate of the regressed parameters is set to $\vec{0}$. The output of the regression network is then fed to the differentiable FLAME decoder network which outputs the 3D head mesh.

The number of ring elements R is a hyper-parameter of our network, which determines the number of images processed in parallel with optimized consistency on the $\boldsymbol{\beta}$. RingNet allows to use any combination of images of the same subject and images of different subjects in parallel. However, without loss of generality, we feed face images of the same identity to $\{e_j\}_{j=1}^{j=R-1}$ and different identity to e_R. Hence for each input training batch, each slice consists of $R-1$ images of the same person and one image of another person (see Fig. 2.3).

2.3.3 Shape consistency loss

For simplicity let us call two subjects who have the same identity label "matched pairs" and two subjects who have different identity labels are "unmatched pairs". A key goal of our work is to make a robust end-to-end trainable network that can produce the same shapes from images of the same subject and different shapes for different subjects. In other words, we want to make our shape generators discriminative. We enforce this by requiring matched pairs to have a distance in shape space that is smaller by a margin, η, than the distance for unmatched pairs. Distance is computed in the space of face shape parameters, which corresponds to

a Euclidean space of vertices in the neutral pose.

In the RingNet structure, e_j and e_k produce $\boldsymbol{\beta}_j$ and $\boldsymbol{\beta}_k$, which are matched pairs when $j \neq k$ and $j, k \neq R$. Similarly e_j and e_R produce $\boldsymbol{\beta}_j$ and $\boldsymbol{\beta}_R$, which are unmatched pairs when $j \neq R$. Our shape constancy term is then

$$\left\|\boldsymbol{\beta}_j - \boldsymbol{\beta}_k\right\|_2^2 + \eta \leq \left\|\boldsymbol{\beta}_j - \boldsymbol{\beta}_R\right\|_2^2 \tag{2.4}$$

Thus we minimize the following loss while training RingNet end-to-end,

$$L_S = \sum_{i=1}^{n_b} \sum_{j,k=1}^{R-1} \max(0, \left\|\boldsymbol{\beta}_{ij} - \boldsymbol{\beta}_{ik}\right\|_2^2 - \left\|\boldsymbol{\beta}_{ij} - \boldsymbol{\beta}_{iR}\right\|_2^2 + \eta) \tag{2.5}$$

which is normalized to,

$$L_{\mathrm{SC}} = \frac{1}{n_b \times R} \times L_S \tag{2.6}$$

where n_b is the batch size for each element in the ring.

2.3.4 2D feature loss

Finally, we compute the L_1 loss between the ground-truth landmarks provided during the training procedure and the predicted landmarks. Note that we do not directly predict 2D landmarks, but 3D meshes with known topology, from which the landmarks are retrieved.

Given the FLAME template mesh, we define for each OpenPose (Simon et al. [2017]) keypoint the corresponding 3D point on the mesh surface. Note that this is the only place where we provide supervision that connects 2D and 3D. This is done only once. While the mouth, nose, eye, and eyebrow keypoints have a fixed corresponding 3D point (referred to as static 3D landmarks), the position of the contour features changes with the head pose (referred to as dynamic 3D landmarks). Similar to (Cao et al. [2014], Tewari et al. [2018]), we model the contour landmarks as dynamically moving with the global head rotation, see Fig. 2.5. To automatically compute this dynamic contour, we rotate the FLAME template between -20 and 40 degrees to the left and right, render the mesh with texture, run OpenPose to predict 2D landmarks, and project these 2D points to the 3D surface. The resulting trajectories are symmetrically transferred between the left and right sides of the face.

During training, RingNet outputs 3D meshes, computes the static and dynamic 3D landmarks for these meshes and projects these into the image plane using the camera parameters predicted in the encoder output. Henceforth we compute the following L_1 loss between the projected landmarks k_{p_i} and the ground-truth 2D

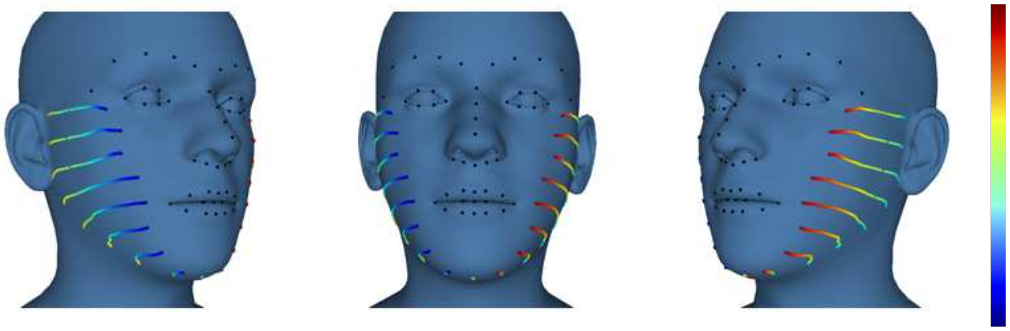


Figure 2.5: **Static and dynamic landmarks:** The figure exhibits static (shown in black) and dynamic (depicted in color) landmark embeddings on the FLAME template. These change based on the viewing angle, with 0° indicating a direct, frontal face perspective. On the right-hand side, a color bar displays the range of angles. The top of this bar corresponds to 40 degrees, which then decreases down the bar. At the bottom of the bar, it signifies angles down to -40 degrees.

landmarks k_i.

$$L_{\text{proj}} = \|w_i \times (k_{p_i} - k_i)\|_1 \tag{2.7}$$

where w_i is the confidence score of each ground-truth landmark which is provided by the 2D landmark predictor. We set it to 1 if the confidence is above 0.41 and to 0 otherwise. The total loss L_{tot}, which trains RingNet end-to-end is

$$L_{tot} = \lambda_{SC} L_{\text{SC}} + \lambda_{proj} L_{\text{proj}} + \lambda_{\boldsymbol{\beta}} \|\boldsymbol{\beta}\|_2^2 + \lambda_{\boldsymbol{\psi}} \|\boldsymbol{\psi}\|_2^2 \tag{2.8}$$

where the λ are the weights of each loss term and the last two terms regularize the shape and expression coefficients. Since $B_S(\boldsymbol{\beta};\mathcal{S})$ and $B_E(\boldsymbol{\psi};\mathcal{E})$ are scaled by the squared variance, the L2 norm of $\boldsymbol{\beta}$ and $\boldsymbol{\psi}$ represent the Mahalanobis distance in the orthogonal shape and expression space.

2.3.5 Implementation details

The feature extractor network uses a pre-trained ResNet-50 (He et al. [2016]) architecture, also optimized during training. The feature extractor network outputs a 2048 dimensional vector. That serves as input to the regression network. The regression network consists of two fully-connected layers of dimension 512 with ReLu activation and dropout, followed by a final linear fully-connected layer with 159-dimensional output. To this 159-dimensional output vector, we concatenate the camera, pose, shape, and expression parameters. The first three elements represent scale and 2D image translation. The following 6 elements are the global rotation and jaw rotation, each in axis-angle representation. The neck and eyeball

Figure 2.6: **Sample images:** Images are taken from NoW dataset.

rotations of FLAME are not regressed since the facial landmarks do not impose any constraints on the neck. The next 100 elements are the shape parameters, followed by 50 expression parameters of FLAME. The differentiable FLAME layer is kept fixed during training. We train RingNet for 10 epochs with a constant learning rate of 1e-4, and use Adam Kingma and Ba [2015b] for optimization. The different model parameters are $R = 6$, $\lambda_{SC} = 1$, $\lambda_{proj} = 60$, $\lambda_{\boldsymbol{\beta}} = 1e-4$, $\lambda_{\boldsymbol{\psi}} = 1e-4$, $\eta = 0.5$. The RingNet architecture is implemented in Tensorflow (Abadi et al. [2016]).We use the VGG2 Face database (Cao et al. [2018]) as our training dataset which consists of face images and their corresponding labels. We run OpenPose (Simon et al. [2017]) on the database and compute 68 landmark points on the face. OpenPose fails in many cases. After cleaning for the failed cases we have around 800K images with their corresponding labels and facial landmarks for our training corpus. We also consider around 3000 extreme pose images with corresponding landmarks provided by Bulat and Tzimiropoulos [2017]. Since for these extreme images we do not have any labels we replicate each image with random crops and scale for matched pair consideration.

2.4 Benchmark dataset and evaluation metric

This section introduces our NoW benchmark for the task of 3D face reconstruction from single monocular images. The goal of this benchmark is to introduce a

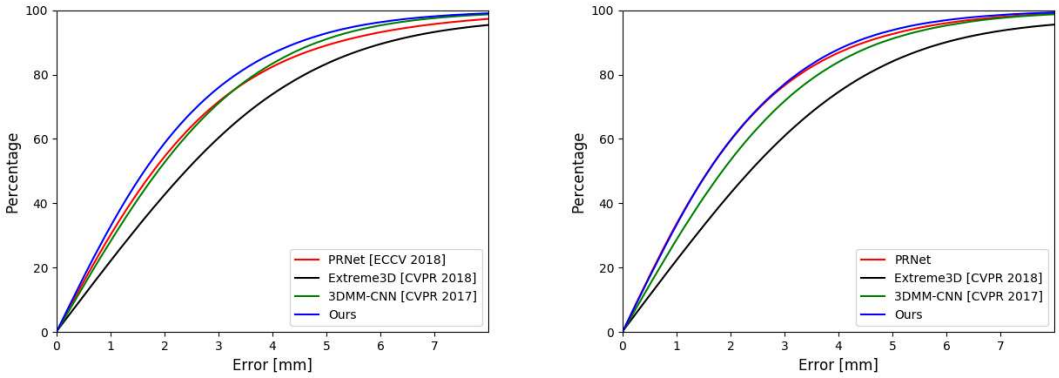


Figure 2.7: **Cumulative error curves:** From left to right: LQ (low resolution) data of Feng et al. [2018b], HQ (High resolution) data of Feng et al. [2018b].

standard evaluation metric to measure the accuracy and robustness of 3D face reconstruction methods under variations in viewing angle, lighting, and common occlusions.

Dataset: The dataset contains 2054 2D images of 100 subjects, captured with an iPhone X, and a separate 3D head scan for each subject. This head scan serves as ground-truth for the evaluation. The subjects are selected to contain variations in age, BMI, and sex (55 female, 45 male).

We categorize the captured data in four challenges; *neutral* (620 images), *expression* (675 images), *occlusion* (528 images) and *selfie* (231 images). *Neutral*, *expression* and *occlusion* contain neutral, expressive, and partially occluded face images of all subjects in multiple views, ranging from frontal view to profile view. *Expression* contains different acted facial expressions such as happiness, sadness, surprise, disgust, and fear. *Occlusion* contain images with varying occlusions from e.g. glasses, sunglasses, facial hair, hats or hoods. For the *selfie* category, participants are asked to take selfies with the iPhone, without imposing constraints on the performed facial expression. The images are captured indoors and outdoors to provide variations of natural and artificial light. Some sample images from the benchmark dataset are shown in Fig. 2.6.

The challenge for all categories is to reconstruct a neutral 3D face given a single monocular image. Note that facial expressions are present in several images, which requires methods to disentangle identity and expression to evaluate the quality of the predicted identity.

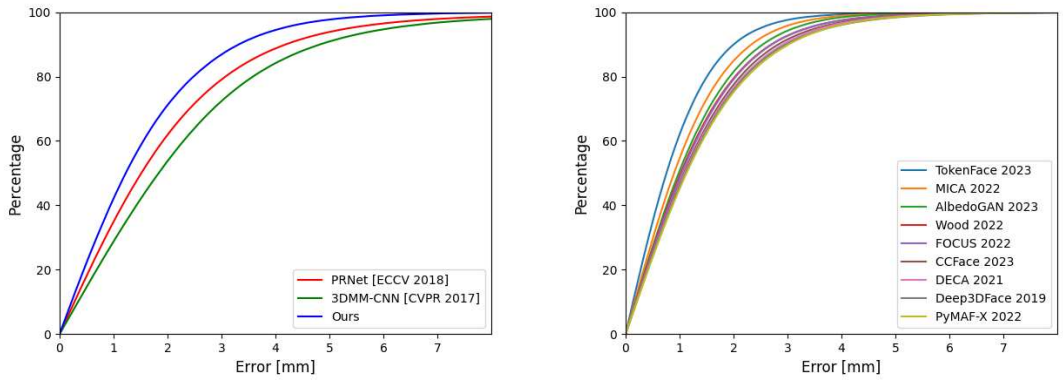


Figure 2.8: **Cumulative error curves:** From left to right: NoW dataset face challenge in 2019 (during the time of publication), NoW dataset face challenge in 2024. Only the top nine methods are shown in the error plot who agreed to make their approach public by 2024. Their rank based on the median error is as follows, TokenFace (Zhang et al. [2023b]), MICA (Zielonka et al. [2022]), AlbedoGAN (Rai et al. [2024]), Wood et al. [2022], FOCUS (Li et al. [2023]), CCFace (Yang et al. [2023]), DECA (Feng et al. [2021]), Deep3DFaceRe Pytorch (Deng et al. [2019b]). For more methods visit the NoW challenge website.

Capture setup: For each subject we capture a raw head scan in a neutral expression with an active stereo system (3dMD LLC, Atlanta). The multi-camera system consists of six gray-scale stereo camera pairs, six color cameras, five speckle pattern projectors, and six white LED panels. The reconstructed 3D geometry contains about 120K vertices for each subject. Each subject wears a hair cap during scanning to avoid occlusions and scanner noise in the face or neck region due to hair.

Data processing: Most existing 3D face reconstruction methods require localization of the face. To mitigate the influence of this pre-processing step we provide for each image, a bounding box, that covers the face. To obtain bounding boxes for all images, we first run a face detector on all images (Zhang et al. [2017]), and then predict keypoints for each detected face (Bulat and Tzimiropoulos [2017]). We manually select 2D landmarks for failure cases. We then expand the bounding box of the landmarks to each side by 5% (bottom), 10% (left and right), and 30% to the top to obtain a box covering the entire face including the forehead. For the face challenge, we follow a processing protocol similar to Feng et al. [2018b]. For each scan, the face center is selected, and the scan is cropped by removing everything outside of a specified radius. The selected radius is subject-specific and

Method	Median (mm)		Mean (mm)		Std (mm)	
	LQ	**HQ**	**LQ**	**HQ**	**LQ**	**HQ**
PRNet (Feng et al. [2018a])	1.79	1.60	2.38	2.06	2.19	1.79
Extreme3D (Trân et al. [2018])	2.40	2.37	3.49	3.58	6.15	6.75
3DMM-CNN (Tuan Tran et al. [2017])	1.88	1.85	2.32	2.29	1.89	1.88
Ours	**1.63**	**1.58**	**2.08**	**2.02**	**1.79**	**1.69**

Table 2.1: Statistics on Feng et al. [2018b] benchmark

is computed as $0.7 \times (outer\_eye\_dist + nose\_dist)$ (see Figure 2.2).

Evaluation metric: Given a single monocular image, the challenge consists of reconstructing a 3D face. Since the predicted meshes occur in different local coordinate systems, the reconstructed 3D mesh is rigidly aligned (rotation, translation, and scaling) to the scan using a set of corresponding landmarks between the prediction and the scan. We further perform a rigid alignment based on the scan-to-mesh distance (which is the absolute distance between each scan vertex and the closest point in the mesh surface) between the ground truth scan, and the reconstructed mesh using the landmark-based alignment as initialization. The error for each image is then computed as the scan-to-mesh distance between the ground truth scan, and the reconstructed mesh. Different errors are then reported including cumulative error plots over all distances, median distance, average distance, and standard deviation.

How to participate: To participate in the challenge, we provide a website `https://now.is.tue.mpg.de/` to download the test images, and to upload the reconstruction results and selected landmarks for each registration. The error metrics are then automatically computed and returned. Note that we do not provide the ground truth scans to prevent users from fine-tuning on the test data.

2.5 Experiments

We evaluate RingNet qualitatively and quantitatively and compare our results with publicly available methods, namely: PRNet (Feng et al. [2018a]), Extreme3D (Trân et al. [2018]) and 3DMM-CNN (Tuan Tran et al. [2017]).

Quantitative evaluation: We compare methods on Feng et al. [2018b] and our NoW dataset.
Feng et al. benchmark: Feng et al. [2018b] describe a benchmark dataset for evaluating 3D face reconstruction from single images. They provide a test dataset,

Method	Median (mm)	Mean (mm)	Std (mm)
PRNet (Feng et al. [2018a])	1.51	1.99	1.90
3DMM-CNN (Tuan Tran et al. [2017])	1.83	2.33	2.05
FLAME-neutral (Li et al. [2017a])	1.24	1.57	1.34
Ours	**1.23**	**1.55**	**1.32**

Table 2.2: Statistics for the NoW dataset face challenge.

that contains facial images and their 3D ground truth face scans corresponding to a subset of the Stirling/ESRC 3D face database. The test dataset contains 2000 2D neutral face images, including 656 high-quality (HQ) and 1344 low-quality (LQ) images. The high quality images are taken in controlled scenarios and the low quality images are extracted from video frames. The data focuses on neutral faces whereas our data has higher variety in expression, occlusion, and lighting as explained in Section 2.4.

Recall that the methods we compare with (PRNet, Extreme3D, 3DMM-CNN) use 3D supervision for training whereas our approach does not. PRNet (Feng et al. [2018a]) requires a very tightly cropped face region to give good results and performs poorly when given the loosely cropped input image that comes with the benchmark database. Rather than try to crop the images for PRNet, we run it on the given images and note when it succeeds: it outputs meshes for 918 of the low-resolution test images and for 509 of high-quality images. To be able to compare with PRNet, we run all the other methods only on the 1427 images for which PRNet succeeds.

We compute the error using the method in Feng et al. [2018b], which computes the distance from ground truth scan points to the estimated mesh surface. Figure 2.7 shows the cumulative error curve for different approaches for the low-quality and high-quality images respectively; RingNet outperforms the other methods. Table 2.1 reports the mean, standard deviation, and median errors.

NoW face challenge: For this challenge we use cropped scans like Feng et al. [2018b] to evaluate different methods. We first perform a rigid alignment of the predicted meshes to the scans for all the compared methods. Then we compute the scan-to-mesh distance (Feng et al. [2018b]) between the predicted meshes and the scans as above. Figure 2.8 shows the cumulative error curves for the different methods; again RingNet outperformed the others at the time of publication. We provide the mean, median, and standard division error (at the time of publication) in Table 2.2.

Qualitative results: Here we show the qualitative results of estimating a 3D face/head mesh from a single face image on CelebA (Liu et al. [2015]) and MultiPIE

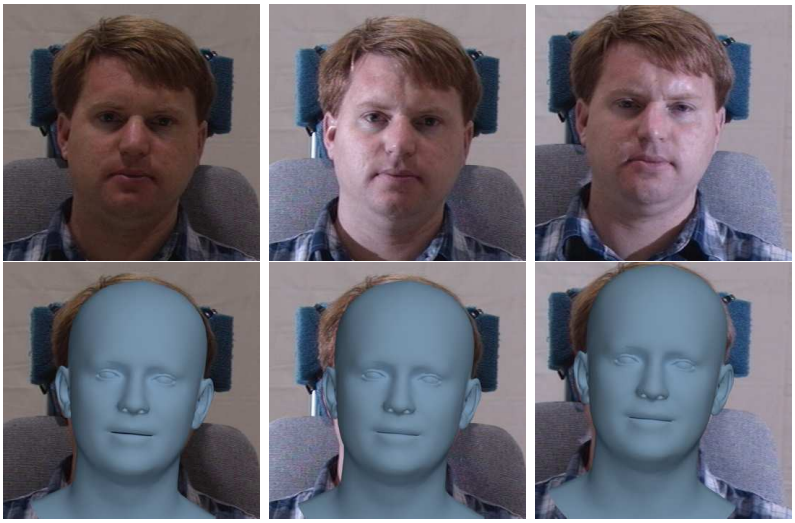


Figure 2.9: **Robustness towards lighting conditions:** Robustness of RingNet to varying lighting conditions. Images from the MultiPIE dataset (Gross et al. [2010]).

dataset (Gross et al. [2010]). Figure 2.1 shows results for RingNet, illustrating its robustness to expression, gender, head pose, hair, occlusions, etc. We show robustness of our approach under different conditions like lighting, poses and occlusion in Figures 2.9 and 2.11. Qualitative comparisons are provided in Fig. 2.10. As can be distinctly seen, RingNet outperforms other methods in terms of facial shape and expression reconstruction under challenging circumstances such as extreme pose, expression, and lighting conditions.

Ablation study: Here we provide the motivation for the choice of using a ring architecture in RingNet by comparing different values for R in Table 2.3. We evaluate these on a validation set that contains 2D images and 3D scans of 10 subjects (six subjects from Dai et al. [2017], four from Li et al. [2017a]). For each subject we choose one neutral scan and two to four scanner images, reconstruct the 3D meshes for the images, and measure the scan-to-mesh reconstruction error after rigid alignments. The error decreases when using a ring structure with more elements over using a single triplet loss only, but it also increases training time. To make a trade of between time and error, we chose $R = 6$ in our experiments.

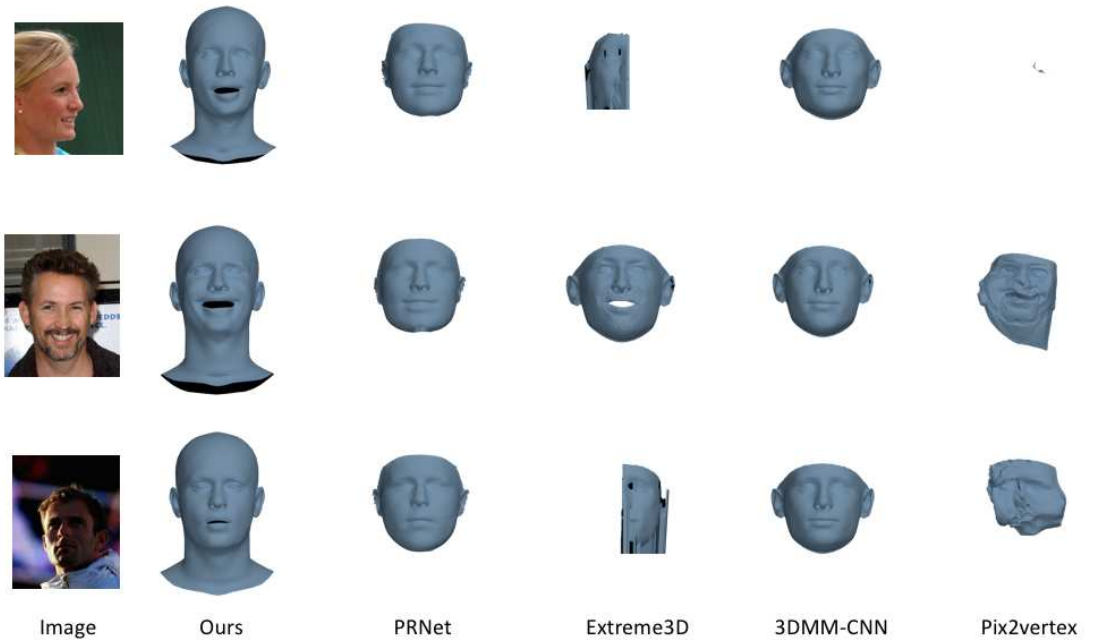


Figure 2.10: **Qualitative comparison:** A qualitative comparison is conducted between RingNet and other methods such as PRNet (Feng et al. [2018a]), Extreme3D (Trân et al. [2018]), 3DMM-CNN (Tuan Tran et al. [2017]), and Pix2vertex (Sela et al. [2017]). It is evident that RingNet delivers both robust and precise facial shapes and expressions in contrast to the other mentioned methods.

2.6 Conclusion

We have addressed the challenging problem of learning to estimate a 3D, articulated, and deformable shape from a single 2D image with no paired 3D training data. We have applied our RingNet model to faces but the formulation is general. The key idea is to exploit a ring of pairwise losses that encourage the solution to share the same shape for images of the same person and a different shape when they differ. We exploit the FLAME face model to factor face pose and expression from shape so that RingNet can constrain the shape while letting the other parameters vary. Our method requires a dataset in which some of the people appear multiple times, as well as 2D facial features, which can be estimated by existing methods. We provide only the relationship between the standard 2D face features and the vertices of the 3D FLAME model. Unlike previous methods we do not optimize a 3DMM to fit 2D features, nor do we use synthetic data. Competing methods typically exploit a photometric loss using an approximate generative model of facial

R	Median (mm)	Mean (mm)	Std (mm)
3	1.25	1.68	1.51
4	1.24	1.67	1.50
5	1.20	1.63	1.48
6	**1.19**	**1.63**	**1.48**

Table 2.3: Effect of varying number of ring elements R. We evaluate on a validation set described in the ablation study.

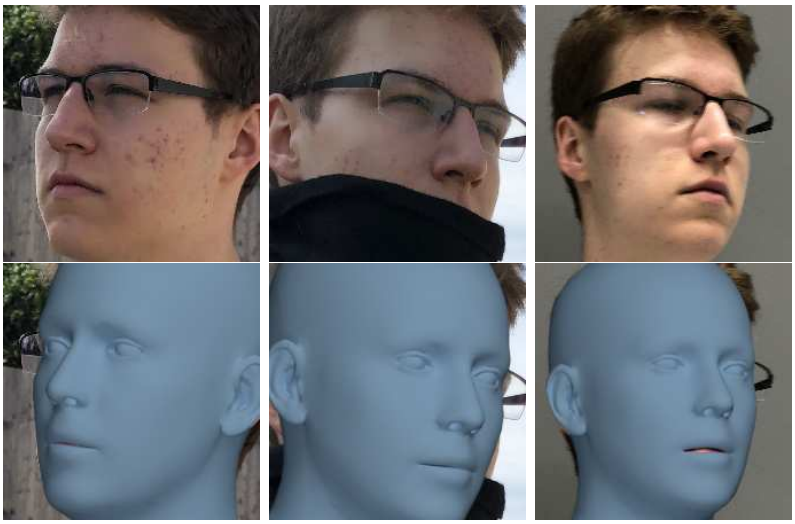


Figure 2.11: **Robustness towards occlusions:** Robustness of RingNet to occlusions, variations in pose, and lighting. Images from the NoW dataset.

albedo, reflectance and shading. RingNet does not need this to learn the relationship between image pixels and 3D shape. In addition, our formulation captures the full head and its pose. Finally, we have created a new public dataset with accurate ground truth 3D head shape and high-quality images taken in a wide range of conditions. Surprisingly, RingNet outperforms methods that use 3D supervision. This opens many directions for future research, for example extending RingNet with Ranjan et al. [2018]. Here we focused on a case with no 3D supervision but we could relax this and use supervision when it is available. We expect that a small amount of supervision would increase accuracy while the large dataset of in-the-wild images provides robustness to illumination, occlusion, etc. Our 2D feature detector does not include the ears, though these are highly distinctive features. Adding 2D ear detections would further improve the 3D head pose and shape. It would be interesting to see if RingNet can be extended to reconstruct 3D body pose and shape from images solely using 2D joints (Pavlakos et al. [2019]). This could go beyond current methods, like HMR (Kanazawa et al. [2018]), to learn

about body shape. While RingNet learns a mapping to an existing 3D model of the face, we could relax this and also optimize over the low-dimensional shape space, enabling us to learn a more detailed shape model from examples. For this, incorporating shading cues (Tewari et al. [2017], Sengupta et al. [2018]) would help constrain the problem.

Chapter 3

Transposition leveraging unpaired data

3.1 Introduction

Now given a single source image of a person, can we generate a realistic image of what they would look like from a different viewpoint, in a different pose, by leveraging unpaired data and informed observations like the previous chapter? While this problem is inherently ambiguous, there is significant statistical regularity in human pose, clothing, and appearance, that could make this possible as illustrated in Fig. 3.1. A solution to the problem would have widespread applications in online fashion, gaming, personal avatar creation or animation, and has consequently generated significant research interest (Dong et al. [2020a], Knoche et al. [2020], Ren et al. [2020], Shi et al. [2020], Song et al. [2019], Yang et al. [2020]).

Recent work focuses on generative modeling (Goodfellow et al. [2014], Isola et al. [2017], Karras et al. [2019], Zhu et al. [2017]), especially using conditional image synthesis. One set of methods uses supervised training (Dong et al. [2020a], Li et al. [2019], Liu et al. [2019], Sarkar et al. [2020]), which requires paired training images of the same person in different poses with the same appearance and clothing. Requiring such paired data limits the potential size of the training set, which can impair robustness and generalization. Consequently, we address this problem *without any paired data* by developing a self-supervised approach similar to the previous chapter. Such self-supervised formulations have also received significant recent attention (Esser et al. [2018], Ma et al. [2018], Pumarola et al. [2018], Yang et al. [2020]).

Our novel formulation builds on the idea of cycle-consistency (Zhu et al. [2017]) with some important modifications. For the forward direction of the cycle, the method takes a source image, source pose and target pose and generates a target image conditioned on pose and appearance. The reverse direction takes this generated image and regenerates the source image by switching the source and target conditions. The goal is to minimize the difference between the original input im-

Figure 3.1: **SPICE** (Self-supervised Person Image CrEation) generates an image of a person in a novel pose given a source image and a target pose. Each triplet in the figure consists of the source image (left), a reference image in target pose (middle) and the generated image in the target pose (right); input and reference images are from the DeepFashion test set (Liu et al. [2016]). Additional results are provided in Appendix C.

age and the one synthesized through the cycle. The problem is that this approach can have a trivial solution in which the cycle produces the identity mapping. To address this, previous methods (Pumarola et al. [2018], Song et al. [2019]) constrain the target image generation with 2D information. Human bodies, however, are non-rigid 3D entities and their deformations and occlusions are not easily expressed in 2D. We show how leveraging 3D information, automatically extracted from images, constrains the model in multiple ways.

Specifically, our method, called SPICE (Self-supervised Person Image CrEation), exploits the estimated 3D body to constrain the image generation, enabling self-supervised learning. In particular, we estimate the SMPL body model (Loper et al. [2015]) parameters corresponding to both the input and the generated target image. Since the input and target image only differ in terms of their pose, their body shape should be the same. SMPL makes this easy to enforce because it factors body shape from pose like FLAME (Li et al. [2017b]). Using this we introduce two losses. First, we use a pose loss that encourages the body pose in the generated image to match the target pose in 3D. Second, we add a shape consistency loss that encourages the person in the generated image to have the same 3D shape as the person in the source image (Fig. 3.2).

These two constraints, however, are not sufficient to generate images with the correct appearance since they only force the model to generate an image with the right shape and pose. There is no constraint that the generated image has the appearance of the source image (e.g. clothing, hair, etc.). Prior work addresses this by enforcing a perceptual loss between patches at each 2D joint (Pumarola et al. [2018]). This is not sufficient when the body is seen with large viewpoint changes or where a body part becomes occluded; see Fig. 3.3. We solve this problem by introducing pose-dependent appearance consistency on the body surface instead of at the joints. The idea is that the projected surface of the 3D body in two different poses must have similar appearance features for matching parts of the body and this similarity should be weighted proportional to the relative global orientation difference between the 3D bodies.

In summary, we improve the realism of self-supervised human reposing by exploiting 3D body information in three novel ways: using a 3D pose loss, body shape consistency, and occlusion-aware appearance feature consistency. We train SPICE with our new constraints on unpaired data. Extensive experiments on the DeepFashion (Liu et al. [2016]) and Fashion Video datasets (Zablotskaia et al. [2019]) show the effectiveness of our model qualitatively and quantitatively. SPICE significantly outperforms the prior state-of-the-art (SOTA) un/self-supervised methods and is nearly as accurate as the best supervised methods.

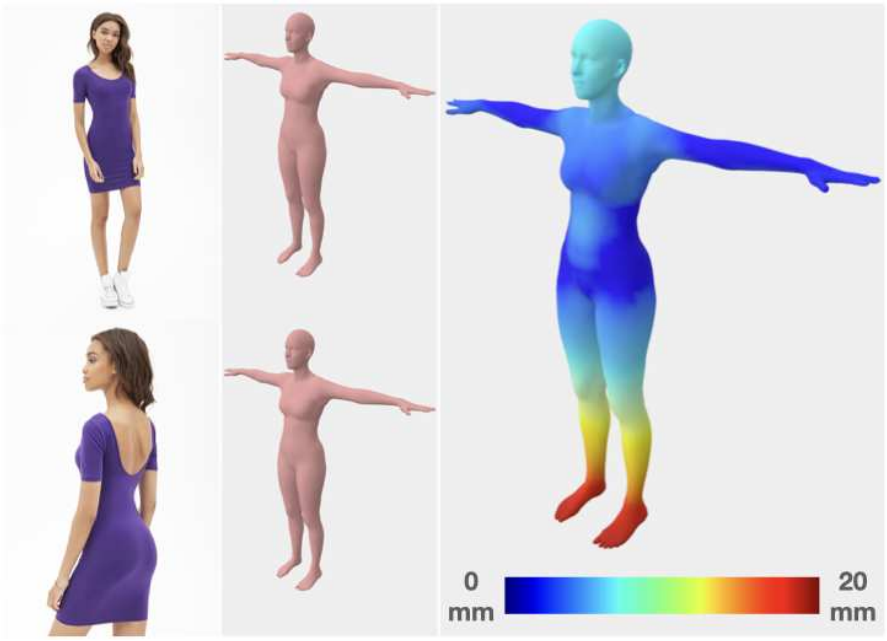


Figure 3.2: **Shape consistency:** The first column shows two images of the same person in two different poses and views. The second column shows the 3D bodies predicted by our 3D regressor and posed in a T-pose. The estimated 3D body shape is similar for the same subject across poses and views. The third column shows the per-vertex difference of both meshes, color coded from blue (0 mm) to red (20 mm).

3.2 Related Work

Methods for reposing images of humans can be broadly divided into two categories: supervised or unsupervised. While both approaches rely on generative modeling (Goodfellow et al. [2014], Isola et al. [2017], Karras et al. [2019], Zhu et al. [2017]), the supervised methods require paired ground truth: source and target training images of the person in different poses. Our approach falls into the unsupervised or self-supervised category, in which we do not use paired training data. We address each class of methods below.

Supervised methods: Supervised approaches learn to transform a source image given a source pose and a target pose (Balakrishnan et al. [2018], Dong et al. [2018, 2020a], Grigorev et al. [2019], Knoche et al. [2020], Li et al. [2019], Liu et al. [2019], Ma et al. [2017, 2020a], Men et al. [2020], Neverova et al. [2018], Ren et al. [2020], Sarkar et al. [2020], Siarohin et al. [2019, 2018a,b], Tang et al. [2020, 2021], Yang et al. [2018, 2021], Zanfir et al. [2020], Zhu et al. [2019], Lakhal et al. [2018]).

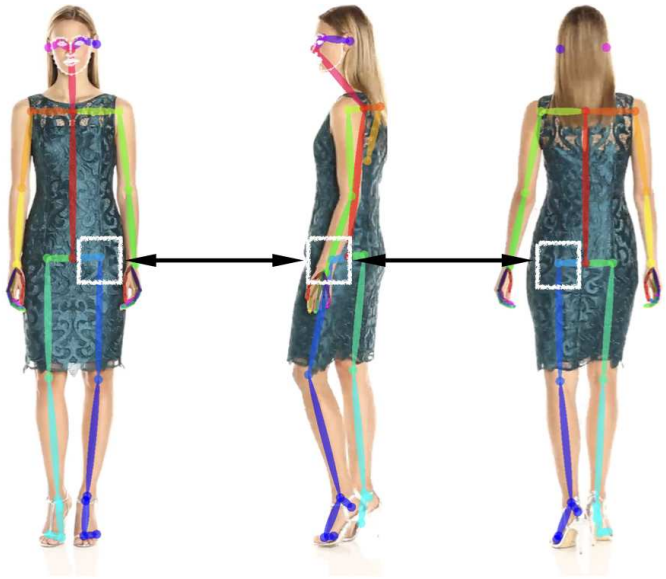


Figure 3.3: **Problem: patch loss based on 2D keypoints.** A person is seen in three different poses with the same clothing. A patch (white rectangle) is extracted at her left hip keypoint. Assuming that the appearance of the patch is the same across viewpoints is incorrect. Instead, SPICE uses the 3D body surface to reason about the regions of the body that are visible in multiple views. Keypoints are predicted by OpenPose (Cao et al. [2019]) for this figure.

Supervision is provided by the target image during training and usually adversarial and perceptual losses are used to train the model (Shi et al. [2020]). The differences between methods usually lie in network inputs and their architectures. Dong et al. (Dong et al. [2018]) synthesize the target image in two stages. First they generate a target pose segmentation from the source pose and use it in their soft-gated warping block architecture to render the person in the target pose. Knoche et al. [2020] learn an implicit volumetric representation of the person to warp the source pose into the target pose. The volumetric representation is implicitly learned using an encoder decoder architecture. Li et al. [2019] utilize a learned flow field to warp a person in a source pose to the target pose. The flow field is learned from 3D bodies and is used for warping at the feature level and pixel level in a deep architecture. Ma et al. [2017] first generate a coarse image of the global structure of a human in the target pose from the source pose in a two stage network. This is then refined in an adversarial way in the second stage to get finer details. Sarkar et al. [2020] compute a partial UV texture map using DensePose (Güler et al. [2018]) from the source image. They use this as input to their network, which learns to complete the UV texture map and render it in a target pose using neural

rendering. Siarohin et al. [2018a] propose a network architecture using deformable skip connections to tackle the problem. Tang et al. [2020] propose a co-attention fusion model that fuses appearance and shape features from images, which they disentangle inside their architecture. They use two different discriminators for appearance and shape to jointly judge the generation. Zhu et al. [2019] propose a progressive generator using a sequence of attention transfer blocks. Each of these blocks transfers certain regions it attends to and generates the image of the person progressively. Ren et al. [2020] propose a new deep architecture where they combine flow-based operations with an attention mechanism. Note that the above methods are all supervised and cannot be directly used in the self-supervised scenario. In contrast, our work is focused on unpaired data and we build on the Ren et al. [2020] architecture to enable this. Thus our contribution does not lie with network architecture but, rather, in introducing novel constraints that make it possible to solve the problem without paired data.

Unsupervised or self-supervised settings: There is an increasing interest in solving the problem in an unsupervised/unpaired manner. Such approaches can work when paired data is not available or can increase robustness and generalization by combining paired and unpaired data. An early approach (Ma et al. [2018]) divides the process in two stages. The first stage uses an auto-encoder-based architecture to learn the corresponding embedding space for pose, foreground and background from source images. The second stage maps Gaussian noise to the embedding space of pose, foreground and background and uses the pretrained decoder from the first stage to generate a person's image in a new pose. Yang et al. [2020] train an appearance encoder from the source image to learn the appearance representation or embedding. They fuse the appearance embedding with the pose embedding coming from an image of a different person in a different pose. In this way they generate the person's image in the new pose. Esser et al. [2018] use a U-Net architecture conditioned on the output of a variational auto-encoder for appearance. They also try to disentangle pose and appearance of a person from the source images.

In general, the mentioned approaches attempt to disentangle shape, pose and appearance in the latent space from a 2D image, which is a hard problem. This results in a relatively poor image generation quality. Instead of learning this disentanglement from images we approach the problem differently. We extract the person's pose and shape information in a parametric decoupled 3D body representation, SMPL (Loper et al. [2015]), and constrain our self-supervised generation. Furthermore, we also constrain appearance generation by leveraging the surface and projection of the 3D body.

Similar to our cyclic formulation, Pumarola et al. (Pumarola et al. [2018]) and

Song et al. (Song et al. [2019]) train their networks in a self-supervised CycleGAN (Zhu et al. [2017]) fashion. Additionally, Song et al. [2019] use semantic parsing maps as input to the network. They constrain their self-supervised generation with 2D information. We differ from these methods by constraining the self-supervised approach with 3D body information.

3.3 Method

SPICE requires a training dataset of tuples (I, P, R), each containing an image I of a person, their pose P as 2D keypoints, and a 2D rendering R. To generate R we fit the SMPL 3D mesh (Loper et al. [2015]) to P using SMPLify (Bogo et al. [2016]), and render the mesh using a color wheel texture in UV space.

We treat all the samples in the dataset as independent; that is, our method does not require images of the same person wearing the same clothing in different poses (i.e. without direct supervision through paired data). During training, the source image I_s, source pose P_s, source rendering R_s, target pose P_t and target rendering R_t are given. SPICE then synthesizes the image $\hat{I}_t$, which is the reposed source image I_s, using a generator network $\mathcal{G}$ (Section 3.3.1):

$$\hat{I}_t = \mathcal{G}(I_s, P_s, R_s, P_t). \tag{3.1}$$

During training, we exploit cycle-consistency (Section 3.3.2). Specifically, we generate a synthetic version of the source image from $\hat{I}_t$ by reusing $\mathcal{G}$; i.e. $\hat{I}_s = \mathcal{G}(\hat{I}_t, P_t, R_t, P_s)$. This enables us to directly apply perceptual and pixel-wise losses between I_s and $\hat{I}_s$ to train $\mathcal{G}$. To prevent trivial solutions, we add 3D guidance (Section 3.3.3) and appearance constraints (Section 3.3.4) for $\hat{I}_t$. See Fig. 3.4 for an overview of the SPICE training pipeline.

3.3.1 Generator architecture

Our generator $\mathcal{G}$ has two modules: a global flow field estimator and a local neural rendering module. The flow estimator module takes R_s, P_s, P_t as input and generates 2D warping fields at the feature level between the source and the target pose. The neural rendering module takes I_s and P_t as inputs and uses the generated warping fields at the feature level of its local attention blocks to generate $\hat{I}_t$. The loss for the flow estimator module can be written as,

$$\mathcal{L}_{flow} = \mathcal{L}_{flow}^{R_s \to R_t} + \mathcal{L}_{flow}^{R_t \to R_s}, \tag{3.2}$$

where $\mathcal{L}_{flow}^{x \to y}$ is the weighted addition of the sampling correctness loss and regularization loss for the generated flow fields, as proposed by Ren et al. [2020]. We refer

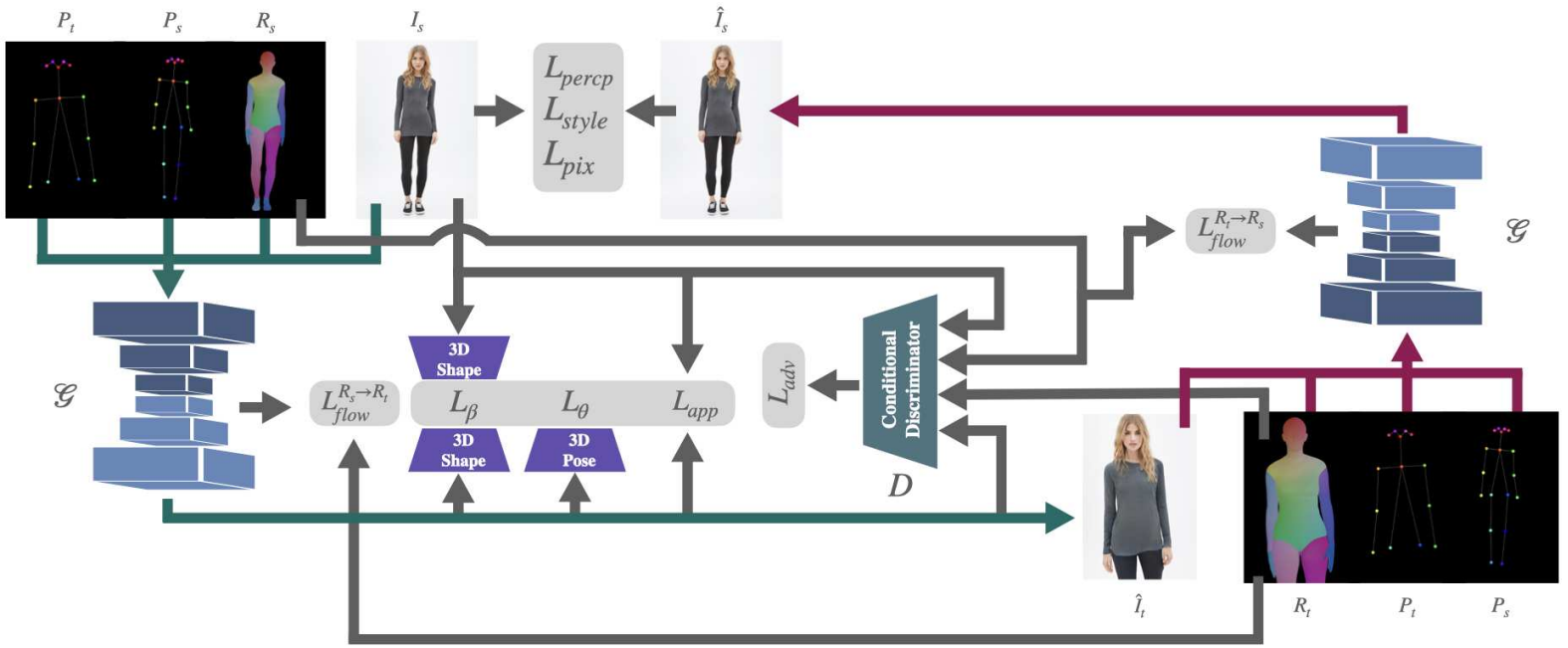


Figure 3.4: **Overview of SPICE:** Given a source image of a person I_s, source pose P_s, target pose P_t and 3D mesh rendering of the source pose R_s, the generator $\mathcal{G}$ generates a target image with the person in the target pose. Then the source and target pose are swapped and passed through $\mathcal{G}$ but with the generated target image as the source. This should re-generate the source image enabling the use of a cyclic self-supervision loss, $\mathcal{L}_{cycle}$, during training. To prevent trivial solutions, the cycle is constrained by losses on 3D pose $\mathcal{L}_{\boldsymbol{\theta}}$, shape $\mathcal{L}_{\boldsymbol{\beta}}$ and appearance $\mathcal{L}_{app}$, which are the main contributions of SPICE (Section 3.3), and an adversarial loss $\mathcal{L}_{adv}$. Note that the P_s and P_t are provided as input heat-maps to $\mathcal{G}$.

to Appendix B for more details on computation of the losses. Here, $\mathcal{L}_{flow}^{R_s \to R_t}$ is applied while synthesizing $\hat{I}_t$, and $\mathcal{L}_{flow}^{R_s \to R_t}$ is applied when regenerating $\hat{I}_s$ at the end of the cycle. The sampling correctness loss is the computed cosine similarity distance between the warped source features and target features. The source and target features come from a specific layer of a pre-trained VGG network (Simonyan and Zisserman [2015]) given the source and target renderings as input, respectively. The regularisation loss provides regularisation to the generated warping fields.

Our generator follows the design of Ren et al. [2020] with the difference that the flow estimator is trained on source and target renderings (i.e. R_s and R_t), instead of source and target images (i.e. I_s and I_t), due to the unavailability of I_t in our setting.

3.3.2 Closing the cycle

Enforcing cycle consistency enables us to train SPICE with supervised losses between the source image I_s and the regenerated source image $\hat{I}_s$. Specifically, we minimize

$$\mathcal{L}_{cycle} = \lambda_{percep}\mathcal{L}_{percep} + \lambda_{style}\mathcal{L}_{style} + \lambda_{pix}\mathcal{L}_{pix}, \tag{3.3}$$

where the λ's are individual loss weights, and the perceptual loss $\mathcal{L}_{percep}$, style loss $\mathcal{L}_{style}$, (Johnson et al. [2016]), and pixel-wise loss $\mathcal{L}_{pix}$ are defined as

$$\mathcal{L}_{percep} = \sum_j \left\| \phi_j(I_s) - \phi_j(\hat{I}_s) \right\|_1$$
$$\mathcal{L}_{style} = \sum_j \left\| \mathbb{G}(\phi_j(I_s)) - \mathbb{G}(\phi_j(\hat{I}_s)) \right\|_1$$
$$\mathcal{L}_{pix} = \left\| I_s - \hat{I}_s \right\|_1,$$

where ϕ_j is the activation map of the j^{th} layer of a pretrained VGG network (Simonyan and Zisserman [2015]), and $\mathbb{G}$ is the Gram matrix built from the activation map ϕ_j.

To generate realistic looking images, SPICE minimizes an adversarial loss by adding a discriminator, D, that discriminates between the fake images $\hat{I}_t$ and real images I_s. To provide pose information along with each image, we condition D on the corresponding rendering (i.e. R_t for $\hat{I}_t$, and R_s for I_s), by providing the concatenation of the two images as discriminator input. Formally, we minimize

$$\mathcal{L}_{adv} = \mathbb{E}[\log(1 - D(\hat{I}_t, R_t))] + \mathbb{E}[\log D(I_s, R_s)]. \tag{3.4}$$

3.3.3 Pose and shape consistency

SPICE uses the SMPL (Loper et al. [2015]) 3D human body model to enforce pose and shape consistency during training. SMPL combines identity-dependent shape blendshapes with pose-dependent corrective blendshapes and linear blendskinning (LBS) for pose articulation. Importantly, this formulation disentangles body shape form pose. Given parameters for shape $\boldsymbol{\beta} \in \mathbb{R}^{|\beta|}$ and pose $\boldsymbol{\theta} \in \mathbb{R}^{3K+3}$, SMPL is a function, $\mathcal{M}(\boldsymbol{\beta}, \boldsymbol{\theta})$ that outputs a 3D mesh with $N = 6890$ vertices.

To extract SMPL shape and pose parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ from I, we use a differentiable regressor (Kolotouros et al. [2019b]), denoted as

$$\boldsymbol{\beta}, \boldsymbol{\theta} = f_{3D}(I). \tag{3.5}$$

Given the extracted SMPL parameters $\hat{\boldsymbol{\beta}}_t, \hat{\boldsymbol{\theta}}_t = f_{3D}(\hat{I}_t)$, we define a loss that encourages the 3D rotation of the joints in the synthetic image, $\hat{\boldsymbol{\theta}}_t$, to be the same as the rotation of the joints in the target pose $\boldsymbol{\theta}_t$:

$$\mathcal{L}_{\boldsymbol{\theta}} = \left\| \boldsymbol{\theta}_t - \hat{\boldsymbol{\theta}}_t \right\|_1, \tag{3.6}$$

where $\boldsymbol{\theta}_t$ is obtained by running SMPLify (Bogo et al. [2016]) on P_t.

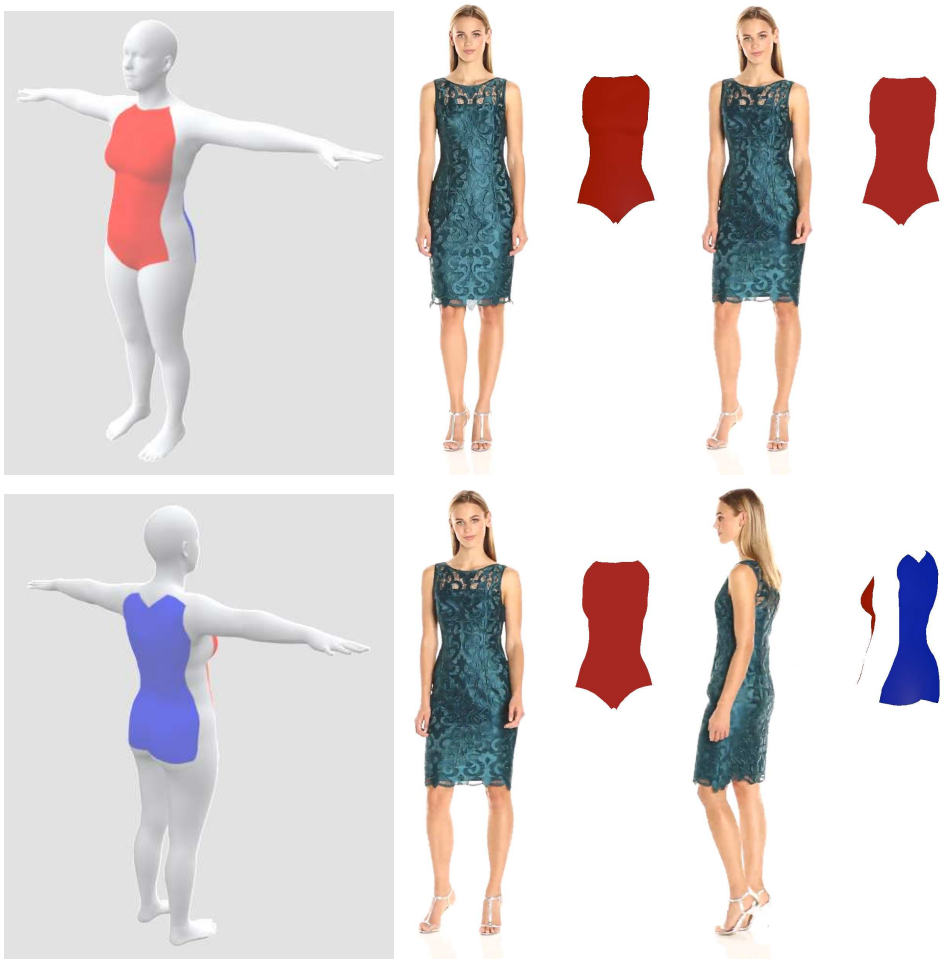
SPICE also enforces body shape consistency (Fig. 3.2) based on the observation that while I_s and $\hat{I}_t$ differ in pose, their body shapes $\boldsymbol{\beta}_s$ (i.e. $\boldsymbol{\beta}_s, \boldsymbol{\theta}_s = f_{3D}(I_s)$) and $\hat{\boldsymbol{\beta}}_t$ must be the same, enforced by

$$\mathcal{L}_{\boldsymbol{\beta}} = \left\| \boldsymbol{\beta}_s - \hat{\boldsymbol{\beta}}_t \right\|_1. \tag{3.7}$$

3.3.4 Appearance feature consistency

The above losses constrain pose and shape in $\hat{I}_t$, but do not guarantee that the appearance of $\hat{I}_t$ remains consistent with I_s. Consequently, we formulate an additional constraint on the appearance of matching regions in I_s and $\hat{I}_t$ to be similar. Due to the unconstrained change in the pose between I_s and $\hat{I}_t$, we cannot apply the deep appearance loss (perceptual or style loss) directly between those images.

Instead, we leverage the 3D body mesh to apply an appearance loss between corresponding image segments. Given SMPL parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$, we render the mesh $\mathcal{M}(\boldsymbol{\beta}, \boldsymbol{\theta})$ with the texture from Fig. 3.5 a) to get the image segments for the rendered front and back torso areas, shown in Fig. 3.5 b), and c). Let M_{mask} denote a binary mask, with value 1 for pixels within the front/back torso segment and 0 elsewhere. Further, let $\mathcal{P}_{patch}$ denote $I \odot M_{mask}$, where $\odot$ is the Hadamard product. Both, m and p are cropped from M_{mask} and $\mathcal{P}_{patch}$, by the bounding box of the image segment.



a) b) c)

Figure 3.5: **Appearance feature consistency:** a) SMPL template with front (red) and back (blue) torso masks. b) and c) show images of a person in different poses (left), and corresponding torso masks obtained by rendering the 3D body with the subject's pose. The appearance consistency loss is then applied on image segments for torso masks of the same color weighted by the relative pelvis rotation.

Given image patches p_s and $\hat{p}_t$ together with binary masks m_s and $\hat{m}_t$, both extracted from I_s and $\hat{I}_t$, the appearance consistency is given as,

$$\begin{aligned} l_{app} = \lambda_{a_1} \sum_k \|\phi_k(p_s) \odot \psi_k(m_s) - \phi_k(\hat{p}_t) \odot \psi_k(\hat{m}_t)\|_1 \\ + \lambda_{a_2} \sum_j \left\| \mathbb{G}_j^{\phi}(p_s) \odot \psi_j(m_s) - \mathbb{G}_j^{\phi}(\hat{p}_t) \odot \psi_j(\hat{m}_t) \right\|_1 \end{aligned} \tag{3.8}$$

where the λ's are weights, ϕ_k is the activation map of the k^{th} layer of pretrained VGG network (Simonyan and Zisserman [2015]), $\mathbb{G}_j^\phi$ is the Gram matrix built from the corresponding activation map ϕ, and ψ is the down-sampling function for the corresponding layer.

Note that the appearance loss, as it is formulated, requires sufficient overlap of corresponding image features within the mask crop. We compute the appearance loss as:

$$\mathcal{L}_{app} = \lambda(\boldsymbol{\theta}_s, \boldsymbol{\theta}_t) \times l_{app}, \tag{3.9}$$

where $\lambda(\boldsymbol{\theta}_s, \boldsymbol{\theta}_t)$ is a weighting function that depends on the relative pelvis rotation (i.e. rotation around the SMPL root joint) between the source and target pose:

$$\lambda(\boldsymbol{\theta}_s, \boldsymbol{\theta}_t) = \begin{cases} 1.0 & \text{if } 0^\circ \leq |\boldsymbol{\theta}_s^{pel} - \boldsymbol{\theta}_t^{pel}| < 20^\circ \\ 0.1 & \text{if } 20^\circ \leq |\boldsymbol{\theta}_s^{pel} - \boldsymbol{\theta}_t^{pel}| < 40^\circ \\ 0.01 & \text{if } 40^\circ \leq |\boldsymbol{\theta}_s^{pel} - \boldsymbol{\theta}_t^{pel}| < 60^\circ \\ 0.0 & \text{otherwise.} \end{cases} \tag{3.10}$$

3.3.5 Final loss

The total loss of the proposed approach is:

$$\begin{aligned} \mathcal{L}_{SPICE} = &\alpha_{cycle}\mathcal{L}_{cycle} + \alpha_{flow}\mathcal{L}_{flow} + \alpha_{adv}\mathcal{L}_{adv} \\ &+ \alpha_{\boldsymbol{\theta}}\mathcal{L}_{\boldsymbol{\theta}} + \alpha_{\boldsymbol{\beta}}\mathcal{L}_{\boldsymbol{\beta}} + \alpha_{app}\mathcal{L}_{app}, \end{aligned} \tag{3.11}$$

where α_i are the corresponding loss weights. The following section provides details on how these weights are set.

3.4 Experiments

Datasets: SPICE is evaluated on two publicly available datasets, namely the DeepFashion In-shop Clothes Retrieval Benchmark (Liu et al. [2016]) and the Fashion Video datatset (Zablotskaia et al. [2019]). The DeepFashion data are used for qualitative and quantitative comparisons and Fashion Video data for motion transfer examples, following Sarkar et al. [2020]. The DeepFashion dataset (Liu et al. [2016]) consists of 52712 high-resolution model images (mostly female) in fashion poses. The data are split into training and testing sets as in previous work (Ren et al. [2020], Zhu et al. [2019]). For training, we use 25341 images from the training set, in which body keypoints from nose to knee are at least visible. Further, 100 randomly selected images from the training set are held out as a validation set for model selection. The qualitative and quantitative evaluations

DeepFashion	Unpaired	FID(↓)	LPIPS(↓)
Def-GAN (Siarohin et al. [2018a])	✗	18.5	0.233
Pose-Attn (Zhu et al. [2019])	✗	20.7	0.253
Intr-Flow (Li et al. [2019])	✗	16.3	0.213
CoCosNet* (Zhang et al. [2020])	✗	14.4	-
ADGAN ** (Men et al. [2020])	✗	22.7	0.183
Ren et al. ** (Ren et al. [2020])	✗	6.4	0.143
VUNet ** (Esser et al. [2018])	✓	34.7	0.212
DPIG ** (Ma et al. [2018])	✓	48.2	0.284
PGSPT ** (Song et al. [2019])	✓	29.9	0.238
SPICE (Ours)	✓	**7.8**	**0.164**

Table 3.1: Quantitative comparison of our method with other state-of-the-art methods. The * denotes that the method reports results for a different train/test split. The ** denotes that the metrics were recalculated using publicly available code and following the protocol described in **Evaluation metrices**; note that recalculation of the metrics results in different numbers from those reported in Ren et al. [2020].

are performed on the same 8570 image pairs as used by Ren et al. [2020]. The Fashion Video dataset (Zablotskaia et al. [2019]) consists of fashion pose video sequences of women in various clothing, captured with a static video camera. The dataset is split into 500 training and 100 test videos as done by Sarkar et al. [2020], with each video containing roughly 350 frames. Please note that SPICE uses no paired images for training.

Training details: We use residual blocks as basic building blocks for $\mathcal{G}$. For more details of the architecture we refer the reader to Ren et al. [2020]. We train SPICE with an image resolution of 256×256 for both datasets. We use spectral normalisation for both the generator and discriminator. The learning rate is 8e$-$4 for $\mathcal{G}$ and 1.6e$-$3 for the discriminator following a GAN training strategy similar to Heusel et al. [2017]. We use 8 NVIDIA V100 GPUs to train SPICE, where each GPU has a batch size of 8. We set the weights for different losses as follows: $\alpha_{cycle} = 1.0, \alpha_{flow} = 1.0, \alpha_{adv} = 1.0, \alpha_{\boldsymbol{\theta}} = 0.01, \alpha_{\boldsymbol{\beta}} = 0.01, \alpha_{app} = 1.0, \lambda_{a_1} = 0.01, \lambda_{a_2} = 10.0, \lambda_{percep} = 0.5, \lambda_{style} = 500.0, \lambda_{pix} = 5.0$. First, we train the flow-field estimator. Differing from Ren et al. [2020], we use R_s and R_t together with keypoints due to the unavailability of I_t during training. R_s and R_t are used as the replacement for I_s and I_t respectively in their flow estimator module. We also finetune the 3D regressor f_{3D} on our training splits of the DeepFashion

dataset (Liu et al. [2016]) following a similar approach of Kolotouros et al. [2019a]. During the finetuning of f_{3D}, we use a representation similar to that proposed by Zhou et al. [2019] for representing 3D rotations. Finally, we train the whole SPICE model end-to-end keeping the 3D regressor weights fixed. During a training iteration we use ROIAlign (He et al. [2017]) to extract the desired regions from I_s and $\hat{I}_t$. We trained our models for 5 days (∼400 epochs). Inference for a single image takes 74 ms using a single NVIDIA V100 GPU.

Evaluation metrics: We use a variety of metrics to evaluate our experimental results in line with the methodology set out in Ren et al. [2020]. These metrics include Learned Perceptual Image Patch Similarity (LPIPS) and Fréchet Inception Distance (FID), as defined by Zhang et al. [2018] and Heusel et al. [2017], respectively. LPIPS measures the perceptual distance between the image created by SPICE and the actual image, thereby assessing the model's reconstruction error. On the other hand, the FID score, defined as the Wasserstein-2 distance between the distributions of real and generated images, measures the realism of images produced by the model.

Further metrics we utilize include contextual similarity (Men et al. [2020]) and object keypoint similarity (OKS) to compare SPICE with other un/self-supervised methods. The former, contextual similarity (CX), assesses the likeness between the source and the image generated by SPICE (CX-GS) as well as between the target and generated image (CX-GT). This is calculated by determining the cosine similarity between deep features extracted from two unaligned images using the VGG19 model. This implementation is drawn from the original codebase (Men et al. [2020], Men).

The latter metric, object keypoint similarity (OKS), measures the distance between the target pose and the pose in the generated image. This is done by using OpenPose (Cao et al. [2019]) to extract the keypoints from the target and the generated images.

For consistency, all images created by the model are padded to a size of 256x256 with a white border. Reference images are prepared by resizing original images from the DeepFashion dataset to a height of 256 and padding them to a size of 256x256 with a white border. We use PyTorch implementations of FID (Seitzer [2020]) and LPIPS (Zhang et al.) which uses AlexNet as a feature extractor. The FID scores are calculated using training images as the reference distribution. The generated images used for the calculation are produced using the test split from Ren et al. [2020] and Zhu et al. [2019].

Quantitative evaluation: Table 3.1 quantitatively compares our method and other state-of-the-art approaches on the DeepFashion dataset (Liu et al. [2016]).

DeepFashion	CX-GS(↑)	CX-GT(↑)	OKS(↑)
VU-Net (Esser et al. [2018])	0.182	0.245	0.93
DPIG (Ma et al. [2018])	0.164	0.197	0.86
PGSPT (Song et al. [2019])	0.169	0.222	0.90
SPICE (Ours)	**0.236**	**0.311**	**0.94**

Table 3.2: Additional quantitative comparison of SPICE with other unpaired state-of-the-art methods.

We compare with Def-GAN (Siarohin et al. [2018a]), Pose-Attn (Zhu et al. [2019]), Intr-Flow (Li et al. [2019]), CoCosNet (Zhang et al. [2020]), ADGAN (Men et al. [2020]), Ren et al. (Ren et al. [2020]), DPIG (Ma et al. [2018]), VUNet (Esser et al. [2018]) and PGSPT (Song et al. [2019]. Note that Def-GAN, Pose-Attn, Intr-Flow, CoCosNet, ADGAN and Ren et al. are supervised methods, requiring the ground-truth image of a person in the target pose and clothing during training. In contrast, our method is unsupervised and comparable with the bottom half of the table (i.e. Ma et al. [2018], Esser et al. [2018]). We have used the publicly available code provided by Ren et al. (Ren et al. [2020]), ADGAN (Men et al. [2020]), VUNet (Esser et al. [2018]), DPIG (Ma et al. [2018]) and PGSPT (Song et al. [2019]) to regenerate the images on our test split and recompute the metrics. SPICE achieves state-of-the-art results among unpaired methods and competitive results when compared with supervised methods. It also significantly outperforms other unsupervised methods in both the contextual similarity scores and the OKS, as demonstrated in Table 3.2.

Qualitative evaluation: Fig. 3.1 shows results on the DeepFashion test split. SPICE does a good job of preserving the target appearance and pose despite the large pose change. Fig. 3.6 provides a qualitative comparison with other un/self-supervised methods on the DeepFashion test split. SPICE generates more realistic and high quality images while preserving pose and appearance compared with DPIG (Ma et al. [2018]), VUNet (Esser et al. [2018]) and PGSPT (Song et al. [2019]). We provide additional qualitative evaluation results in Appendix C.

Motion transfer: If you can generate one pose, you can generate a sequence of poses. Consequently, we show video generation on the test split of the Fashion Video dataset in Fig 3.7. We randomly select one video from the test split to act as a driving video that provides the target pose. We take the first frame of the other videos from the test split as the source image and generate the whole sequence from these. Please note that we did not train SPICE to generate videos;

Figure 3.6: **Qualitative comparison:** We qualitatively evaluate SPICE against other methods that use either unsupervised or self-supervised approaches. The methods we compare it against include DPIG (Ma et al. [2018]), VUNet (Esser et al. [2018]) and PGSPT (Song et al. [2019]). SPICE outperforms these methods by producing images of superior realism and quality. Additionally, it maintains pose and appearance. Additional results are included in Appendix C.

i.e. there is no video supervision or temporal consistency.

Ablation study: Table 3.3 summarizes our ablation study, which removes one loss at a time from the model. The configuration "SPICE w unconditional D" means that we give the generated image to the discriminator without conditioning on pose by concatenating the renderings. Our full model better preserves details, pose, and has better overall image quality. Trained without the pose loss, the generator has less information about the self-occlusions of the body. Therefore it tends to generate poses that are not possible for a real person, e.g. growing legs inside another leg, etc. If we train SPICE excluding the shape loss, the generator has less information about the 3D body shape of the person in the source image,

Figure 3.7: **Qualitative results** on the Fashion Video dataset (Zablotskaia et al. [2019]). Video frames are synthesized from the source frame using the poses in the driving video.

Configuration	FID(↓)	LPIPS(↓)
SPICE w/o shape loss	8.7	0.166
SPICE w/o pose loss	8.4	0.165
SPICE w/o appearance loss	9.9	0.164
SPICE w unconditional D	10.0	0.167
SPICE	7.8	0.164

Table 3.3: Ablation study on DeepFashion test set Liu et al. [2016].

GEN \ REF	80	90	95	RAW
80	6.9	7.3	8.1	12.1
90	7.5	7.1	7.4	10.6
95	8.7	7.8	7.4	9.6
RAW	12.4	10.4	9.1	**7.8**

Table 3.4: FID as a function of the JPEG quality level for generated (GEN) and reference images (REF).

which can lead to inconsistent deformations of shape in the generated images; e.g. having bigger hips with a very thin waist, etc. Excluding the appearance loss during training leads to less detailed reconstructions and an overall reduced clothing consistency. Fig. 3.8 illustrates such loss-specific artifacts.

FID for different JPEG quality levels: The common practice to calculate metrics on generated images is to save the images on disk in JPEG format as an intermediate step. We noticed that this affects the FID calculation significantly, as shown in Table 3.4. The Frechet Inception Distance (FID) score increases with disparities in JPEG compression ratios between real and generated images. Moreover, the FID score decreases when higher levels of JPEG compression are consistently applied to both the real and generated image distributions.

Discussions and limitations: While the DeepFashion dataset (Liu et al. [2016]) provides paired data, these pairs do not always have the same outfit, as can be seen in the bottom row of fig. 3.6. A manual inspection of 500 randomly selected pairs from the training dataset revealed that 16% of these pairs exhibit discrepancies, with one image in each pair featuring additional accessories or altered clothing. Supervised learning methodologies necessitate the use of identical subjects, maintaining consistent attire and accessories, across two distinct poses

Figure 3.8: **Loss specific artifacts:** Each row shows artifacts when training without a specific loss. Top: without shape loss. Middle: without pose loss. Bottom: without appearance loss. From left to right: source image, reference image in the target pose, generated without the corresponding loss, and SPICE, respectively.

Figure 3.9: **Limitations:** SPICE has difficulty super-resolving fine details when zooming, dealing with extreme closeups, and generating humans from clothing images without humans.

for effective training. Variations in attire or the introduction of new accessories within these poses can affect the training efficiency of these methods. Instead, we take the extreme approach of pure self-supervision to see how far this can be pushed. For extreme pose/view changes, the solution is highly ambiguous: there is no way to know the front of an outfit from the back or vice versa. Although SPICE generates a plausible solution, the result might not match the real invisible details. A practical use case would limit the range of pose variation between source and target. SPICE requires the target pose of a person where much of the body is in view (Fig. 3.9). Our model has difficulties preserving fine patterns when the camera zooms in Fig. 3.9. Zoom requires super-resolution which is a research topic in itself.

3.5 Conclusion

We have presented SPICE, a novel approach to repose clothed humans from a single image. SPICE is trained in a self-supervised fashion without paired training data by exploiting cycle consistency. Our key insight is to use 3D body information during training in different ways to constrain the image generation. First, SPICE leverages a parametric 3D body model and a 3D body regressor to constrain body shape and pose. Second, SPICE uses the 3D body mesh to coherently segment source and generated images to enforce an occlusion-aware appearance feature

consistency. Third, SPICE conditions a discriminator on colored mesh renderings to increase the quality of the generated images. Once trained, SPICE takes a single image and a target pose specified by 2D keypoints, and generates an image of the same person in the target pose. SPICE generates images that are significantly better than previous unsupervised methods, and that are similar in quality to the state-of-the-art supervised method at the time of publication. Additionally, SPICE can readily generate videos, although it is not trained for this task.

Adding 3D constraints to the reposing problem enables a number of applications that go beyond the scope of this and belong to future work. Although we used our shape and appearance losses to keep those traits constant, they could as well be used to control the output model appearance (e.g. changing the pattern of a T-shirt) or shape (e.g. changing the body proportions of the model).

Chapter 4

Generation leveraging unpaired data

4.1 Introduction

Generating 3D virtual humans that can be articulated with realistically deforming clothing, is a key challenge of content creation for games and movies. Virtual assistants in augmented and virtual reality could be enriched by an automatically generated 3D human appearance. Furthermore, synthetic humans could also play an important role in data generation to comply with privacy and data protection laws. 3D scenes with moving humans can be synthesized to create datasets for training, for example, human pose estimation. In the previous chapters, we discussed the modeling of geometry and appearance of digital human avatars, addressing each aspect separately; Chapter 2 concentrated on reconstructing geometry, while Chapter 3 dealt with the appearance to enable controllable digital humans. The question now arises: can we use the informed insights gained from data like these chapters to jointly model both the geometry and appearance, thereby enhancing the control over digital human generation?

Recently, we have seen immense progress in the synthesis of virtual 3D humans (Hong et al. [2023], Zhang et al. [2022], Grigorev et al. [2021], Bergman et al. [2022], Noguchi et al. [2022], Jiang et al. [2022]). However, they are almost exclusively based on implicit representations, such as neural radiance fields (Noguchi et al. [2022], Hong et al. [2023]). These implicit representations are incompatible with classical rendering frameworks, making it challenging to integrate them into existing applications. To address this, we propose SCULPT a generative model of explicit 3D geometry (meshes) and the appearance (texture maps) of clothed humans (Fig. 4.1). There are several challenges that we need to address to build such a model. To naively train the 3D generative model, a large dataset of 3D scanned humans would be required containing a diverse set of people in a variety of different poses, wearing different outfits. Not only such data is not publicly available, but it would also be very expensive to collect. However, we observed that

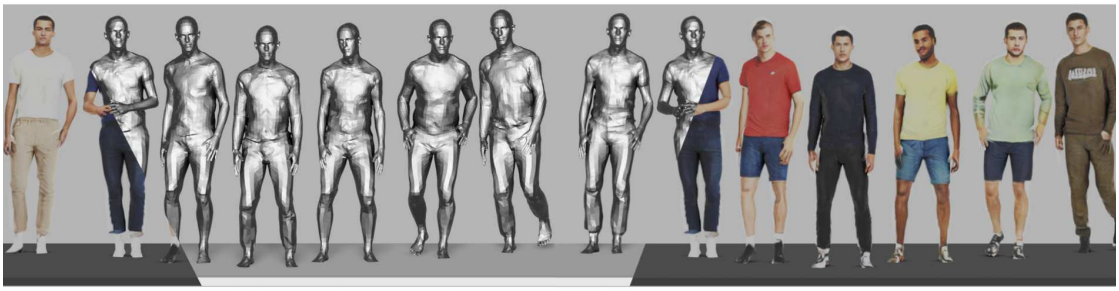


Figure 4.1: **SCULPT** is a generative model of geometry and appearance of clothed human meshes. The generated textured clothing mesh can be readily inserted into 3D scenes. In the above figure, we generated the clothed humans and placed them on a 3D floor. The scene has a single camera with global directional light settings. Additional results are included in Appendix D.

there are different requirements for learning the geometry and texture of a generative model. For geometry learning, there are medium-scale datasets of 3D-scanned humans that are publicly accessible. Additionally, 2D images depicting humans in various outfits are abundantly available, which can assist in learning appearance or texture. It is also noted that clothing items, such as t-shirts, despite having identical geometries, can vary significantly in colors and patterns. This diversity allows for a wide range of colors and textures to be associated with similar geometrical structures. Based on this observation, we propose to leverage medium-scale datasets of 3D scans to learn the geometry distribution, while large-scale 2D image datasets are used to learn the appearance model. Based on these data sources, and leveraging the foundational SMPL body model (Loper et al. [2015]), we design our explicit generative model.

Specifically, we modify the StyleGAN architecture to output (i) pose-dependent geometry in terms of displacement maps, and (ii) geometry-dependent appearance in terms of a color texture, in the texture space (UV-space) of the SMPL template. Our geometry model is trained using the CAPE dataset (Ma et al. [2020b]), which contains pose annotations as well as registered SMPL meshes. We use the trained geometry model to condition our texture generator, which is trained on a large collection of 2D fashion images. Note that we train the texture model in an unsupervised way, only relying on adversarial losses, thereby avoiding the requirement of 3D data paired with 2D images. The process of geometry conditioning plays a crucial role in maintaining coherence between appearance and geometry. In essence, this process ensures that the visual and geometric attributes of the clothed human harmoniously conform to each other. To mitigate dependence of generated clothing type or its color on the body pose, we condition both the tex-

ture and the geometry generators with attribute labels. This approach reduces the entanglement between pose and clothing attributes, resulting in more realistic and accurate generation. We automatically generate these labels by using the visual question-answering model BLIP (Li et al. [2022]), and CLIP (Radford et al. [2021]).

During inference, our model has the ability to generate a diverse set of clothed, textured 3D humans that can be controlled by various parameters such as clothing type, and clothing appearance such as color. This level of control over the generated output is a significant advantage of our approach, as it allows for great flexibility in generating realistically clothed and textured 3D humans. The coarse clothing color can be controlled from text-based inputs, which enables users to specify the desired color for a given piece of clothing without requiring detailed knowledge of 3D modeling or design. Moreover, since our generative model produces 3D meshes it is compatible with existing graphics and game engines, allowing seamless integration into a range of applications. In summary, the contribution of this work is a novel hybrid learning strategy for unsupervised and unpaired learning of a generative model of 3D virtual humans. It is enabled by coupling the appearance and geometry network via language-driven attribute labels.

4.2 Related Work

Generative 3D modeling: 3D-aware generative models have been receiving a tremendous amount of focus from the computer vision community in recent years (Chan et al. [2022], Or-El et al. [2022], Gao et al. [2022], Chan et al. [2021], Schwarz et al. [2020, 2022]). This has resulted in different representations of 3D objects and scenes like implicit functions and different rendering techniques such as volumetric rendering. In the following paragraph, we provide a brief overview of these concepts as they help contextualize our current work better.

Chan et al. [2021] propose a new architecture for generative models, where they build the generator with SIREN networks (Sitzmann et al. [2020]) and perform volumetric rendering to generate 2D images. The SIREN network models the inherent geometry of an image using implicit representations which are rendered to a 2D image via volumetric rendering inspired by NeRF (Mildenhall et al. [2020]). Or-El et al. [2022] combine an SDF-based 3D representation with a style-based 2D generator. The SDF-based representation helps in achieving geometry details but produces low-resolution image features, which are then fed to a 2D-based style generator that produces high-resolution images. Chan et al. [2022] take an interesting direction by decoupling feature generation and neural rendering with the help of a tri-plane representation. This enables them to leverage the power of superior image generation modules like StyleGAN2 (Karras et al. [2020b]) to

generate high-quality 2D results. Even though these works have advanced the state-of-the-art in 3D aware image synthesis for relatively simple objects like cars or faces, their generated geometry is not of high quality. Moreover, they are not directly usable for complex 3D deformable and articulable models like clothed humans. This requires incorporating more refined domain information.

This leads to a more focused set of generative models (Hong et al. [2023], Zhang et al. [2022], Grigorev et al. [2021], Bergman et al. [2022], Noguchi et al. [2022], Jiang et al. [2022], Ma et al. [2020b], Chen et al. [2022], Corona et al. [2021], Palafox et al. [2021]) which are concentrated on generative 3D humans. Among these 3D generative models, a stream of work (Ma et al. [2020b], Chen et al. [2022], Corona et al. [2021], Palafox et al. [2021]) uses 3D data as supervision. Yet they only model geometry and not the texture or appearance of the clothed humans. This is caused by the limited size of 3D datasets containing the large variation of clothing textures along with their corresponding geometry. To get around this limitation, another stream of work uses 2D images as supervision. The concurrent works by Bergman et al. [2022], Zhang et al. [2022]) and Noguchi et al. [2022] extend Chan et al. [2022] for human bodies where they learn a neural radiance field for a canonical pose that is later reposed. The radiance fields are learned by using a similar concept of tri-planes proposed by (Chan et al. [2022]). The final human image produced by the network of Bergman et al. [2022] is blurry, however, and the geometry is very smooth and lacks deformations relating to clothing. Similarly, Noguchi et al. [2022] also produce blurry results and undesired artifacts. Grigorev et al. [2021] propose to generate neural textures using 2D generative models. The neural texture is superimposed on minimally clothed SMPL-X (Pavlakos et al. [2019]) meshes and rendered. The rendered images are then passed via a neural renderer to generate realistic-looking 2D person images. Their method does not alter the geometry and is incompatible with existing infrastructure, *e.g.*, game engines. Hong et al. [2023] divide the whole body into parts and learn local radiance fields, which are then integrated to get the final rendered image. They also suffer from undesirable artifacts and lack any control over geometry and texture. In contrast, we learn geometry on top of a fixed topology mesh (SMPL, Loper et al. [2015]) and the corresponding texture map for the underlying mesh. This is done via learning from unpaired data using a novel training approach utilizing both 3D and 2D data. Displacements on top of SMPL are sufficient for many kinds of clothes. Moreover, this template is readily compatible with any existing 3D rendering engine. Additionally, unlike prior art, we provide explicit controls over the clothing type and color.

Generative 2D humans: If we set aside the goal of having an explicit 3D mesh as output, some 2D methods become relevant. Here, we study a few generative

models that can generate high-quality 2D images for clothed humans (Sarkar et al. [2021b], Fu et al. [2022]). Fu et al. [2022] propose a curated dataset of fashion images and train StyleGAN (Karras et al. [2021, 2020b]). In this way, they produce high-quality 2D images of clothed humans but lack controllability on the generated pose and other aspects, such as global orientation, clothing type, etc. Sarkar et al. [2021b] propose a new architecture for controllable human generation. To provide additional pose controllability over the generated humans one can use any of the 2D reposing algorithms (Balakrishnan et al. [2018], Dong et al. [2018, 2020b], Grigorev et al. [2019], Knoche et al. [2020], Esser et al. [2018], Pumarola et al. [2018], Yang et al. [2020], Sanyal et al. [2021], Ma et al. [2018], Pumarola et al. [2018], Sanyal et al. [2021], Song et al. [2019], Sarkar et al. [2021a], AlBahar et al. [2021]) proposed in the recent years. In summary, these methods propose various architectures and/or training procedures that enable them in synthesizing a human in a new pose given the image of the person in a different pose. Although these methods generate high-quality images, they have no notion of the underlying geometry. Therefore they cannot be used in classical 3D graphics pipelines or AR and VR applications.

4.3 Method

SCULPT is a generative model that takes a geometry code $\mathbf{z}_g \sim \mathcal{N}(\mathbf{0}, I^{512\times512})$, a texture code $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, I^{512\times512})$, body pose $\boldsymbol{\theta} \in \mathbb{R}^{69}$, clothing geometry type $\mathbf{c}_g \in \{0,1\}^6$, and clothing texture description $\mathbf{c}_t \in \mathbb{R}^{512}$ as input, and it generates a clothed 3D body mesh $\mathcal{M} := \{V, \mathcal{C}\}$ with a texture image $\mathcal{I}_{tex} \in \mathbb{R}^{256\times256\times3}$ (Fig. 4.2). Here, $V \in \mathbb{R}^{6890\times3}$ represents a set of 6890 3D vertices, and $\mathcal{C}$ is a fixed set of triangles represented as a 3-tuple of vertex indices in SMPL (Loper et al. [2015]) mesh topology. Formally, SCULPT is defined as:

$$\mathcal{M}, \mathcal{I}_{tex} = SCULPT(\mathbf{z}_g, \mathbf{z}_t, \boldsymbol{\theta}, \mathbf{c}_g, \mathbf{c}_t). \tag{4.1}$$

SCULPT models clothing geometry as vertex displacements from the minimally clothed SMPL body in the canonical pose. To account for pose articulation effects, the clothing generator is conditioned on pose. The texture generator in turn takes the features from the geometry generator as a conditioning signal. Therefore, the output of SCULPT can be fully articulated with SMPL's pose control, and the textured mesh is readily usable in existing graphics applications.

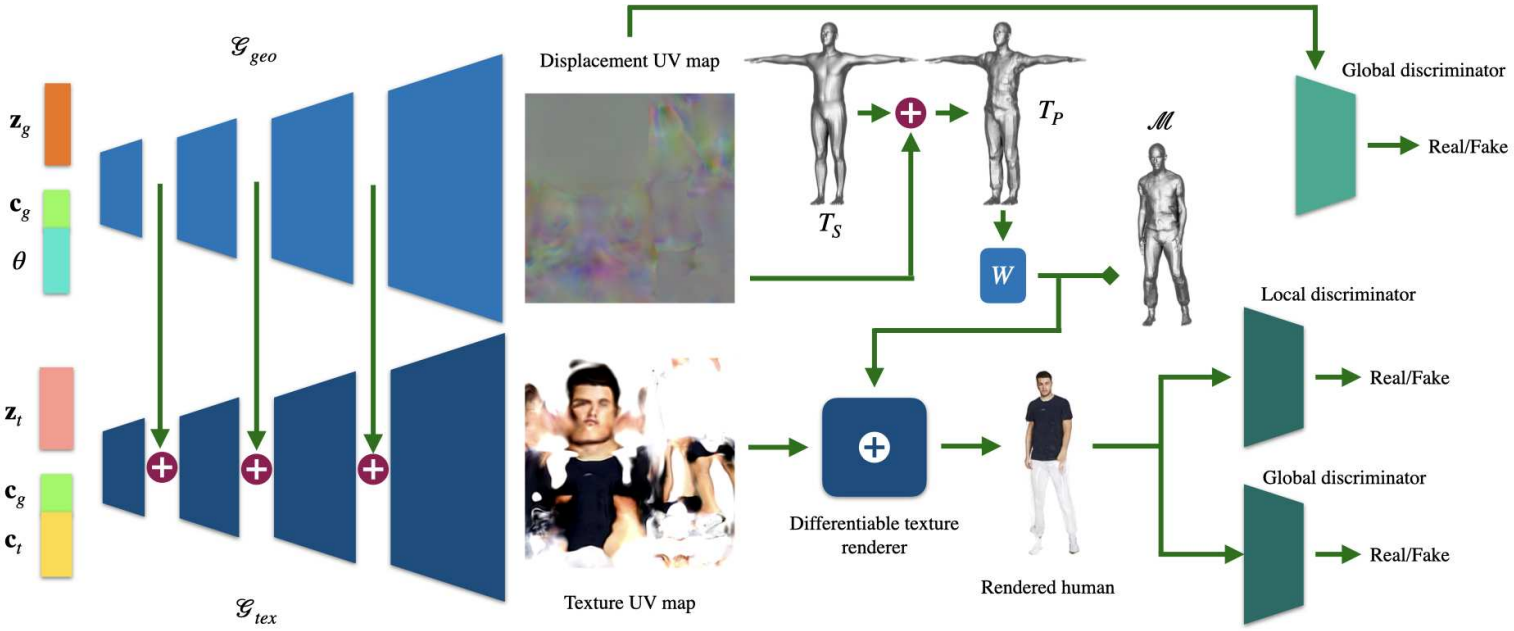


Figure 4.2: **Overview:** SCULPT consists of two StyleGAN-based generators for geometry ($\mathcal{G}_{geo}$) and appearance ($\mathcal{G}_{tex}$), both acting in the UV space of the SMPL body model. The geometry network $\mathcal{G}_{geo}$ outputs pose-dependent displacement maps that are added to the SMPL template mesh and is trained using 3D scan data. Based on this model, the appearance generator $\mathcal{G}_{tex}$ is trained in an unsupervised way using adversarial losses computed on rendered images of the generated synthetic human. It is conditioned on intermediate features of the geometry network. Besides the noise code, both generator networks receive additional attributes for appearance ($\mathbf{c}_t$) and clothing type ($\mathbf{c}_g$) as input. This enhances the connection between appearance and geometry, and it offers a user-friendly control over the generation.

4.3.1 Clothing representation

SCULPT adapts the SMPL (Loper et al. [2015]) model formulation to clothed bodies with additional parameters $\mathbf{z}_g$ and $\mathbf{c}_g$ as:

$$T_P(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}_g, \mathbf{c}_g) = T_S(\boldsymbol{\beta}) + B_P(\boldsymbol{\theta}) + V_{geo}(\mathbf{z}_g, \mathbf{c}_g, \boldsymbol{\theta}), \tag{4.2}$$

where $T_S(\boldsymbol{\beta}) \in \mathbb{R}^{6890\times 3}$ denotes the SMPL body in "canonical pose" for the shape parameters $\boldsymbol{\beta} \in \mathbb{R}^{10}$. $B_P(\boldsymbol{\theta}; \mathcal{P}) : \mathbb{R}^{3K} \rightarrow \mathbb{R}^{6890\times 3}$ are the SMPL pose corrective blend shapes, and V_{geo} are the pose-dependent clothing vertex displacements. The vertex displacements V_{geo} are obtained by sampling the UV displacement map output by the clothing geometry generator $\mathcal{G}_{geo}$ (see Section 4.3.2) at fixed UV coordinates for every vertex.

As the clothing geometry is defined as offsets from the SMPL body in the canonical pose, it can be reposed as:

$$\mathcal{M}(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}_g, \mathbf{c}_g) = W(T_P(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}_g, \mathbf{c}_g), \mathbf{J}(\boldsymbol{\beta}), \boldsymbol{\theta}), \tag{4.3}$$

where $W(T_P, \mathbf{J}, \boldsymbol{\theta})$ is SMPL's blend skinning function, which rotates the vertices of T_P around $K = 23$ joints $\mathbf{J} \in \mathbb{R}^{3K}$. The joint locations $\mathbf{J}$ are defined as a function of the body shape.

4.3.2 Clothing geometry generator

Given a random geometry code $\mathbf{z}_g \sim \mathcal{N}(\mathbf{0}, I^{512\times 512})$, a one-hot clothing type vector $\mathbf{c}_g \in \{0,1\}^6$, and the SMPL body pose $\boldsymbol{\theta}$, the clothing geometry generator outputs a UV displacement map $UV_{geo} \in \mathbb{R}^{256\times 256\times 3}$ (see left column of Figure 4.2). Formally, the generator is defined as:

$$UV_{geo} = \mathcal{G}_{geo}(\mathbf{z}_g | \mathbf{c}_g, \boldsymbol{\theta}). \tag{4.4}$$

Following CAPE (Ma et al. [2020b]), $\mathbf{c}_g$ are categorical classifications of the clothing type like "short sleeve T/shirt-short trouser", "short sleeve T/shirt-long trouser", "long sleeve T-shirt/long trouser", "long sleeve T-shirt/short trouser", "shirt/long trouser", "shirt/short trouser", provided with the CAPE dataset, and which are represented as one-hot vectors. Intuitively, $\mathbf{c}_g$ controls the clothing category, while $\mathbf{z}_g$ and $\boldsymbol{\theta}$ model clothing variations and pose-dependent deformations within the particular clothing category, respectively (Fig. 4.3). To sample clothed 3D body meshes from SCULPT, we sample vertex displacements V_{geo} from the generated displacement map UV_{geo} and evaluate Equation (4.3).

The generator $\mathcal{G}_{geo}$ follows a StyleGAN3 (Karras et al. [2021]) architecture, which is trained with an adversarial loss from a global discriminator. We compute

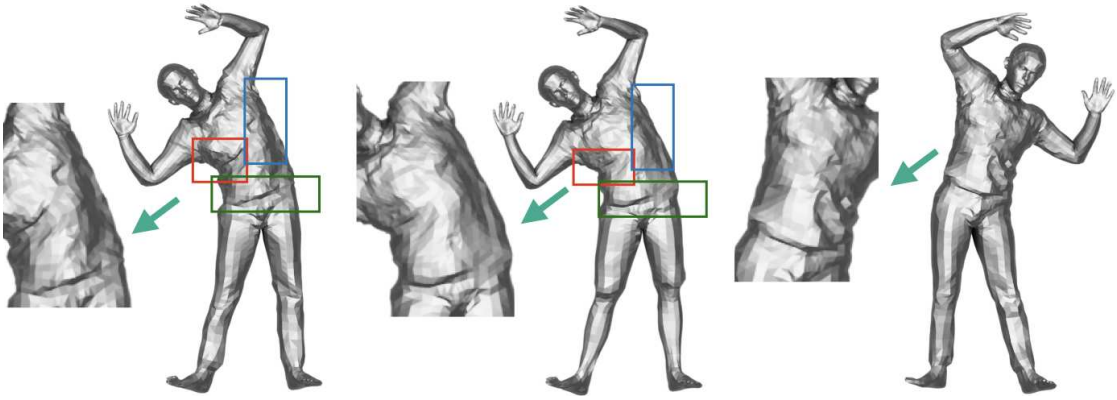


Figure 4.3: **Pose control:** The figure presents three meshes of an individual. The initial two meshes depict the individual in different types of clothing (long-long and short-short) but maintaining the same pose, while the third mesh illustrates the individual in a different pose, wearing the same type of clothing as depicted in the first mesh. The pose-dependent clothing deformations are visible in the zoomed-in images on the side, which are produced by the geometry generator. It is observed that identical poses result in similar deformations, as indicated by the color-coded bounding boxes, while a change in pose leads to distinct clothing deformations in the geometry.

the displacement maps from the scan registrations provided by the CAPE dataset, which are treated as real samples for the discriminator. For the fake examples, we combine random noise vectors for $\mathbf{z}_g$ and $\mathbf{c}_g$ with randomly sampled CAPE data poses. During training, the output of the generator is masked by the segmentation mask of the SMPL UV map and passed to the discriminator as fake samples. Following standard conditional GAN training, the discriminator is also given $\mathbf{c}_g$ and $\boldsymbol{\theta}$ as inputs.

4.3.3 Texture generator

Given a random texture code $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, I^{512\times512})$, our texture generator $\mathcal{G}_{tex}$ generates a UV texture image. To control the generated texture $\mathcal{G}_{tex}$, it is conditioned on clothing texture descriptors $\mathbf{c}_t$. Additionally we condition the texture generator with the categorical clothing type $\mathbf{c}_g$ and intermediate features of the clothing geometry generator $\mathcal{G}_{geo}$. Formally, the texture generator is:

$$\mathcal{I}_{tex} = \mathcal{G}_{tex}(\mathbf{z}_t|\mathbf{c}_g, \mathbf{c}_t, \mathcal{G}_{geo}). \tag{4.5}$$

We train the texture generator from a collection of 2D fashion images, obtained

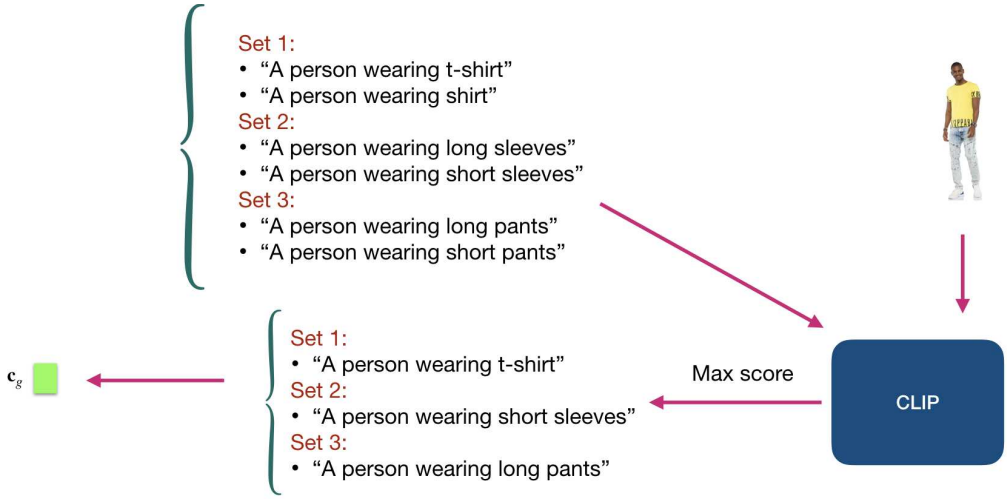


Figure 4.4: **Computation of $\mathbf{c}_g$ for fashion images:** Given an image and three different prompt sets, we use CLIP to assign scores to each image-text pair, and select the top matching pair per set. The results from these three sets are amalgamated into a single categorical label that describes the clothing type featured in the fashion image. This label is analogous to the clothing-type labels used by the geometry generator.

from fashion websites. For each training image, as detailed in section 4.3.4, we automatically extract clothing color descriptors $\mathbf{c}_t$ and clothing types $\mathbf{c}_g$ with the help of a visual question answering system – BLIP (Li et al. [2022]) and CLIP (Radford et al. [2021]). Realistic clothing appearance, however, consists of more than coarse clothing colors. Namely, it can contain varying color patterns. We capture such variations in our generator using an additional latent vector $\mathbf{z}_t$. The coarse clothing geometry type $\mathbf{c}_g$ and the clothing color descriptor $\mathbf{c}_t$ are concatenated, and provided as input condition to $\mathcal{G}_{tex}$. This is sufficient to generate good visual quality texture but the generated texture is inconsistent with the generated geometry. For instance, a short-sleeved shirt covers a smaller skin region of the arms as compared to a long-sleeved one, therefore the texture generator must be conditioned using the geometry information. Although the clothing category $\mathbf{c}_g$ loosely correlates with the generated texture and geometry, it is not enough. For instance, without explicit information on the clothing part boundaries (e.g. boundary between shirt and trousers, or boundary between cloth and skin) the texture network is unable to generate a clothing appearance that conforms to the clothing geometry. To account for the correlation of generated geometry and texture, we condition $\mathcal{G}_{tex}$ on intermediate features of $\mathcal{G}_{geo}$. Specifically, during a forward pass, $\mathcal{G}_{geo}$ takes the batch of $\mathbf{z}_g, \mathbf{c}_g$, and $\boldsymbol{\theta}$, and it generates features at different synthesis blocks.

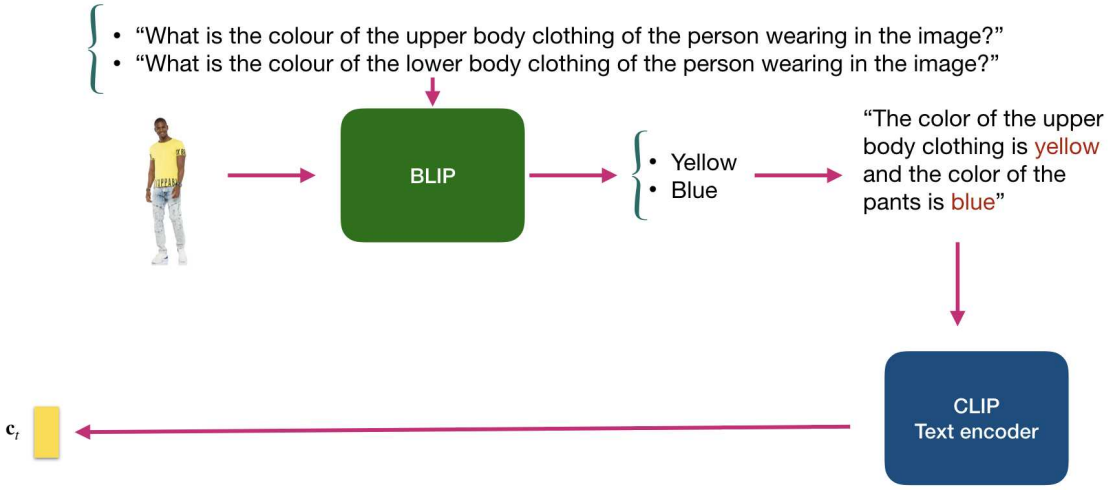


Figure 4.5: **Computation of $\mathbf{c}_t$ for fashion images:** A fashion image along with two distinct questionsare sequentially inputted into the visual question answering model, BLIP. BLIP's output is structured into a sentence which is subsequently fed into the CLIP text encoder to extract a feature vector which is $\mathbf{c}_t$. During the inference process, the sentence is formulated using user-created color inputs, replacing the role of the BLIP model.

Since $\mathcal{G}_{tex}$ and $\mathcal{G}_{geo}$ share StyleGAN's model architecture, we progressively add at each level, excluding the mapping network, the feature blocks of $\mathcal{G}_{geo}$ to the feature blocks of $\mathcal{G}_{tex}$. See the left half of Fig. 4.2 for a visual representation of this technique. This effectively passes the signals from the geometry network to the texture generator.

Finally, we combine the generated $\mathcal{I}_{tex}$ and $\mathcal{M}$, and render the textured mesh with a differentiable texture renderer (Ravi et al. [2020]). We use a global and a patch-based discriminator simultaneously to train $\mathcal{G}_{tex}$. While the global discriminator acts on the whole rendered clothed human image, the patch-based discriminator is a local discriminator that acts on the random 64×64 patches extracted from the image. Following the conditional GAN training scheme, $\mathbf{c}_g$ and $\mathbf{c}_t$ are provided as inputs to both the discriminators. Empirically, we find that the local discriminator helps improve the image quality of the body parts, whereas the global one ensures consistency of the entire structure.

4.3.4 Obtaining clothing texture descriptions

We utilize the language-to-image models CLIP (Radford et al. [2021]) and BLIP (Li et al. [2022]) to automatically label the fashion images with clothing type $\mathbf{c}_g$ and clothing color descriptors $\mathbf{c}_t$. Specifically, to compute, $\mathbf{c}_g$ we pass one fashion

image through CLIP (Radford et al. [2021]) and compute its image features. Then we use augmented prompts for the categorical clothing types, e.g., "the person is wearing a t-shirt", "the person is wearing a shirt", "the person is wearing long pants" etc. as input to CLIP's text feature extractor. Finally, we use the scores provided by CLIP for each text input and the corresponding fashion image and select the features corresponding to the text prompt with the highest score as its clothing type descriptor $\mathbf{c}_g$. Please refer to Fig. 4.4 for a more visual description of this process. To compute the color descriptor $\mathbf{c}_t$, we pass the fashion images to BLIP (Li et al. [2022]) and query its visual question answering (VQA) model with questions such as, "What is the color of the upper body clothing of the person wearing in the image?". BLIP then outputs a textual description of the clothing color. We augment this text with the following sentence, "The color of the upper body clothing is [BLIP output text] and the color of the pants is [BLIP output text]". This is then passed as text input to CLIP to get the text-based feature, which then serves as $\mathbf{c}_t$. At inference time, we replace the BLIP-generated output with a fixed textual description of the clothing colors, and use this as input to CLIP, to compute $\mathbf{c}_t$. Please refer to Fig. 4.5 for a more visual description of this process.

4.3.5 Training and dataset details

SCULPT is implemented in PyTorch (Paszke et al. [2019]) and optimized with ADAM (Kingma and Ba [2015a]) with a learning rate of 0.001. The geometry, texture generators and the discriminators follow the StyleGAN3-t (Karras et al. [2021]) architecture. The fake examples for the texture discriminators of $\mathcal{G}_{tex}$ are obtained by rendering the generated clothed body mesh $\mathcal{M}$ with the generated texture map $\mathcal{I}_{tex}$ using PyTorch3D (Ravi et al. [2020]). All generators and discriminators are trained using a non-saturating GAN loss using R1 regularisation (Karras et al. [2020b], Mescheder et al. [2018]) and the adaptive augmentation technique from StyleGAN-Ada (Karras et al. [2020a]).

Our model utilizes PyTorch3D's soft rasterizer for differentiable rendering, with a zero blur radius for one-to-one pixel-triangle correspondence. A directional light with fixed intensity and orthographic projection is employed for mesh rendering. Body orientation or view for each rendered image is randomly chosen from the training dataset, with this randomization applied per image in each batch during generator forward passes.

The training process follows two stages: (1) The geometry generator is trained on the CAPE dataset (Ma et al. [2020b]) until the FID converges. The CAPE dataset (Ma et al. [2020b]) consists of SMPL registrations to the scans of 41 subjects wearing different types of clothing. The dataset comes with six different clothing type variations, namely "short-short", "short-long", "long-long", "long-

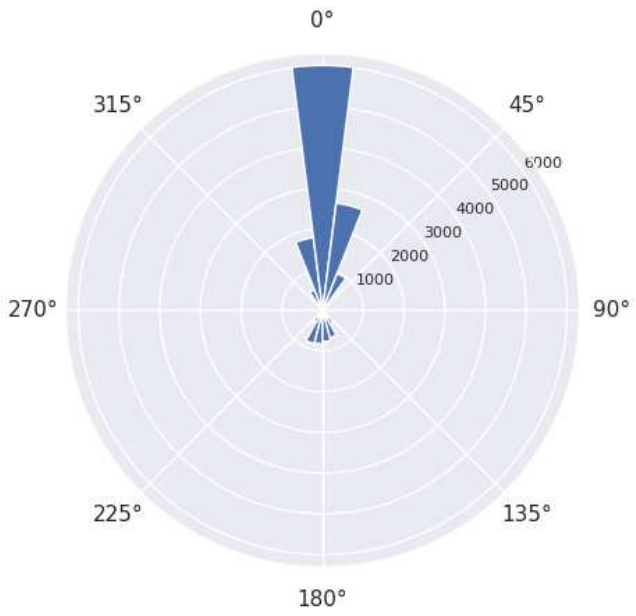


Figure 4.6: Histogram of the body rotations of the used training corpus with respect to the camera view (0° is frontal).

short", "shirt-long" and "short-short". Here, the term "long-short" represents that the person is wearing a round neck shirt/t-shirt with long sleeves and short pants. Similarly, the term "short-long" represents that the person is wearing a round-neck shirt/t-shirt with short sleeves and long pants. The same terminology is followed for the other labels. We encode these six clothing types as a 6-dimensional one-hot vector. Each registered mesh is unposed, i.e., effects of pose articulation and translation are removed, and the vertex offsets from the minimally-clothed SMPL body are represented in the UV space as a displacement map. In total, we compute 63069 displacement maps from these registered meshes. (2) Then the geometry model is kept fixed and used for training the texture generator until the FID converges.

The texture generator is trained on a curated dataset of fashion images obtained from the catalog images uploaded in the website of Zalando (zal [2021]). We collected 16362 fashion images, normalized the images (human-centered in the middle), and removed the background. Refer to Fig. 4.6 for the view statistics of our collected dataset. It is observed that the dataset exhibits a bias towards both frontal and near-frontal views. The dataset offers a variety of clothing types, including 'short sleeve T-shirt/short trouser", "short sleeve T-shirt/long trouser", "long sleeve T-shirt/long trouser", "long sleeve T-shirt/short trouser", "shirt/long

trouser", and "shirt/short trouser" which are similar to the assortment found in the CAPE dataset. These labels are automatically computed using CLIP, as detailed in Section 4.3.2. The dataset contains 2,483 "short-short", 6,260 "short-long", 335 "long-short", 3,425 "long-long", 939 "shirt-short", and 2,920 "shirt-long" items, where "short-short" refers to "short sleeve T-shirt/short trousers", and so forth. Regarding the color types in the training dataset of fashion images, there are descriptions for 115 different colors. Examples of these colors include red, blue, green, khaki, pink, peach, and tan, among others. We plan to release the dataset annotations for research purposes.

We run MODNet (Ke et al. [2022]) on the aligned images to get the segmentation masks for the foreground body and use the pose regressors provided by ICON (Xiu et al. [2022]) to estimate the SMPL pose and shape of the bodies. We compute the clothing color information utilizing CLIP and BLIP as described in Section 4.3.4. In a perceptual study with 2000 labeled images on Amazon Mechanical Turk, BLIP labels were judged to be correct 92.7% of the time for upper-body clothing and 89.7% for lower-body clothing. During the study, participants received images alongside associated BLIP labels and assessed their validity with a "yes/no" answer. We treat human judgments as ground truth. The common point of mismatch between the participants and BLIP labels was nearby colors like khaki or tan etc.

4.4 Experiments

We conduct a three-part evaluation of SCULPT. Initially, we assess its controllability properties. Subsequently, we compare it with state-of-the-art methods. Finally, we execute a detailed series of ablation studies.

Controllability of SCULPT: In Fig. 4.7, we show generation results by varying the body pose $\boldsymbol{\theta}$, while keeping all other control parameters of SCULPT fixed. This leads to pose-dependent deformation of the clothing geometry, which is visible in the side-by-side comparison of textured and textureless mesh for each pose-pair of one identity. Additionally, we show the pose-dependent deformations only in the geometry in Fig. 4.3. Notice that the appearance of the clothes changes in tandem with the geometry. In Fig. 4.8, we vary the clothing type, $\mathbf{c}_g$ keeping all other parameters fixed. As intended, the clothing type changes from long sleeves and long pants to short sleeves and long pants when $\mathbf{c}_g$ is changed accordingly (Row 1, identity 2). Fig. 4.9 shows that for each clothing geometry, we can generate different colored garments by varying $\mathbf{c}_t$, keeping the other factors fixed.

Our model offers further fine-grained control over the appearance and allows for fine changes in the texture of the clothing, i.e., it can generate different shades of

Figure 4.7: **Pose control:** Varying pose while keeping other factors fixed. Each row contains two different identities, each in two poses. Texture and geometry meshes are shown side-by-side.

the same coarse color, and different patterns on the same t-shirt/shirt, etc. This is achieved by varying $\mathbf{z}_{tex}$ in Fig. 4.10. It is worth noting that varying $\mathbf{z}_{tex}$ changes the texture patterns for identity 1 in row 1 of Fig. 4.10 and it generates different shades of the same color. In contrast to 2D generative modeling, our output is a textured mesh that can be rendered under arbitrary viewpoints, see fig. 4.11.

Comparisons with SOTA: We compare SCULPT with two state-of-the-art (SOTA) methods, EG3D (Chan et al. [2022]) and EVA3D (Hong et al. [2023]) quantitatively in Table 4.1 and qualitatively in Fig. 4.12. We also provide qualitative comparisons of SCULPT with two additional SOTA methods, namely GET3D (Gao et al. [2022]) and StylePeople (Grigorev et al. [2021]) in Fig. 4.13. We compare to these methods only qualitatively because an official implementation of training code of StylePeople (Grigorev et al. [2021]) is not publicly available and GET3D (Gao et al. [2022]) requires images of people in a canonical pose to train, which is not available for our dataset.

We find that although SCULPT and EG3D both methods generate high-quality images. However, the underlying geometry, generated by EG3D, of EG3D is of low quality. We hypothesize that the highly articulated human body is significantly

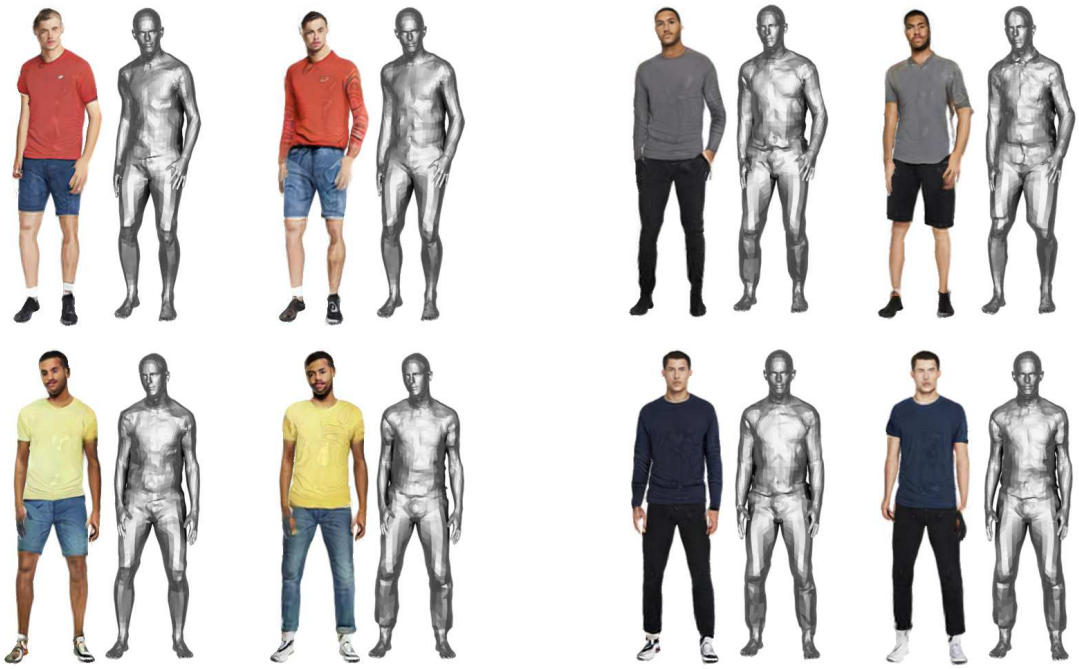


Figure 4.8: **Cloth-type control:** Varying $\mathbf{c}_g$ while keeping other factors fixed. Each row contains two separate identities. For each identity, we show two different clothing types consecutively. Texture and geometry meshes are shown side-by-side.

more complex to model as compared to human and animal faces, which have less articulation. This renders the training of EG3D difficult, leading to undesirable solutions. Moreover, EG3D does not provide any of the highly desirable control parameters that our model provides.

Recently, EVA3D (Hong et al. [2023]) has been proposed to add controllable articulation to 3D aware generative models of the human body. For a fair comparison, we used EVA3D's publicly available implementation and DeepFashion experiment parameters and trained it with our 256X256 texture data. Note that the original EVA3D model was trained on a different dataset with 512x512 resolution, leading to different results. By doing so, EVA3D improves the quality of the generated geometry as compared to EG3D. But their generated texture is of lower quality than SCULPT which is evident from Table 4.1 and Fig. 4.12. Furthermore, as the generated geometry is represented as an implicit function, using it in existing graphics engines requires converting it into meshes, which is time-consuming and often does not preserve the rendering quality.

While GET3D's geometric representation can model loose-fitting clothes such as skirts, it lacks articulation and pose control (it generates the body in a canonical

Figure 4.9: **Cloth-color control:** $\mathbf{c}_t$ is varied while keeping other factors fixed. Each row consists of two different clothing geometries and differently colored garments for that geometry.

pose). While SCULPT is more limited in topology due to the explicit representation, it enables control over complex, articulated human figures, and generates better textures (Fig. 4.13). Adopting different explicit topologies for different clothing types could greatly enhance the types of clothes SCULPT can model. This is left for future work.

In contrast to StylePeople, which employs 2D neural rendering with generated neural textures, our approach estimates accurate clothing geometry that is compatible with standard 3D renderers. Contrary to StylePeople, SCULPT incorporates a geometry branch. The feature outputs from each Style-Block in the geometry generator are added with those from the corresponding ones in the texture generator, resulting in a texture that is consistent with the generated geometry. This fundamental difference in our architecture allows us to generate better-quality renders compared to StylePeople as can be seen in Fig. 4.13. In summary, SCULPT outperforms the SOTA methods in terms of geometric representation (StylePeople), generated geometry quality (EG3D, EVA3D, StylePeople), articulation (GET3D, EG3D), texture quality (all), and final human image production (all) (Fig. 4.12 and Fig. 4.13).

Figure 4.10: **Cloth-texture fine control:** Varying $\mathbf{z}_{tex}$ while keeping other factors fixed. Each row consists of two different clothing geometries, each with textures generated for the same color condition but with different $\mathbf{z}_{tex}$.

Ablation experiments: To better understand the contribution of different components in SCULPT, we perform ablation experiments as shown in Table 4.2. These experiments involve altering the choice of discriminator combinations, patch sizes, and the conditioning of the texture network by the intermediate activations of the geometry network. We train the geometry network conditioned texture network with only one discriminator at a time in cases (b), (c), and (d). However, we observe that the global discriminator alone is not capable of generating sharp results (b). On the other hand, the patch discriminators work relatively well in improving local parts of the body but lack global correspondence. Among the local discriminators, we compare the performance of two patch sizes, 64×64 (d) and 32×32 (c), and find that 64×64 performs better. We hypothesize that as the granularity of the local discriminator increases, its field of view diminishes, leading to a higher likelihood of encountering white backgrounds devoid of human elements, particularly in comparison to a 64×64 patch. This phenomenon potentially impairs the model's overall performance. Ganokratanaa et al. [2020] observed a similar pattern, where their discriminator using 64×64 patches outperformed the one with 32×32 patches. We then add the global discriminator along with the local discriminator in cases (e) and (f), which improves the overall perfor-

Figure 4.11: **Viewpoint changes:** Rotating the textured mesh.

mance. The 64×64 patch size also performs best in the combined discriminator strategy as can be seen by comparing (c) and (e), and (d) and (f). Finally, we build a baseline where we train the texture generator without conditioning from the geometry network with the dual discriminator strategy, as shown in (a). However, the baseline perform poorly compared to our full model (f), as the geometry and texture do not conform to each other. We further demonstrate in Fig. 4.14 that the geometry network's conditioning of the texture network allows the texture to conform to the clothing geometry, as observed in the clothing boundaries and wrinkles of the different identities.

Method	FID ↓	KID ↓	Precision/Recall ↑
EG3D	7.38	0.0036	0.79/0.14
EVA3D	44.11	0.0387	0.30/0.03
SCULPT	9.85	0.0063	0.53/0.22

Table 4.1: **Quantitative comparison:** We evaluate our model using the standard FID, KID, and precision and recall (Sajjadi et al. [2018]) metrics against the most recently proposed similar methods. Our rendering quality is comparable to that of state-of-the-art (SOTA) methods.

Figure 4.12: **Qualitative comparison:** We compare with EG3D (left) and EVA3D (middle). Our rendered humans (right) have comparable quality with EG3D whereas our geometry surpasses both. More comparisons are available in Figure D.10 and Figure D.11.

Limitations and discussion: Similar to existing generative clothed human body models, SCULPT generates poor-quality textures in the backside region because of a dataset bias towards frontal and near-frontal views, and to "typical" fashion poses in the fashion image dataset used for training. The dataset bias towards viewpoint can be observed from the dataset statistics shown in Fig. 4.6. The quality also degrades for challenging, unseen body poses Fig. 4.15. The existing datasets further lack diversity in age, race, skin tone, and gender. The predominance of male examples is due to the CAPE dataset bias towards tight clothing and the prevalence of male subjects in our texture training data. To overcome the limitation in view- and pose diversity, one can train our model on multi-view datasets or videos of subjects in the same clothing but in varying poses and visible from different viewpoints.

SCULPT sometimes generates hand textures with the same color as the clothing. The issue partly arises from about 30% of the training set's fashion images showing hands in pockets, as observed from manual image examination of a hundred images.

Figure 4.13: **Additional qualitative comparisons.** From left to right: StylePeople (Grigorev et al. [2021]), GET3D (Gao et al. [2022]), SCULPT (each with two results). Images are taken directly from the respective publications.

Ablated	(a)	(b)	(c)	(d)	(e)	(f)
FID ↓	28.2	31.6	24.5	16.1	19.2	9.85

Table 4.2: **Ablation study:** (a) full model without geometry conditioning; (b) full model trained with only global discriminator; (c) full model trained with only local discriminator of patch size 32×32; (d) full model trained with only local discriminator of patch size 64×64; (e) full model trained with both global and local discriminator of patch size 32×32; (f) full model trained with both global and local discriminator of patch size 64×64.

However, the CAPE dataset used for geometry training lacks hands-in-pockets instances, leading to model ambiguity in recognizing hand positions in pockets. Augmenting the dataset with annotations specifying hand positions could mitigate this.

Finally, our model also has topological limitations in modeling loosely fitting clothes such as skirts or long dresses. To handle loosely fitting clothing such as skirts, our method would require a corresponding mesh template that offers the correct topology. Given the limited range of typical clothing typologies, it could be feasible to design a few different typologies to accommodate such clothing types but this is outside of the scope of this work. Instead, we focus on developing a novel method for learning geometry and texture without paired 3D training data. We use SMPL as a template for our explicit representation (mesh) which is compatible with current graphics engines.

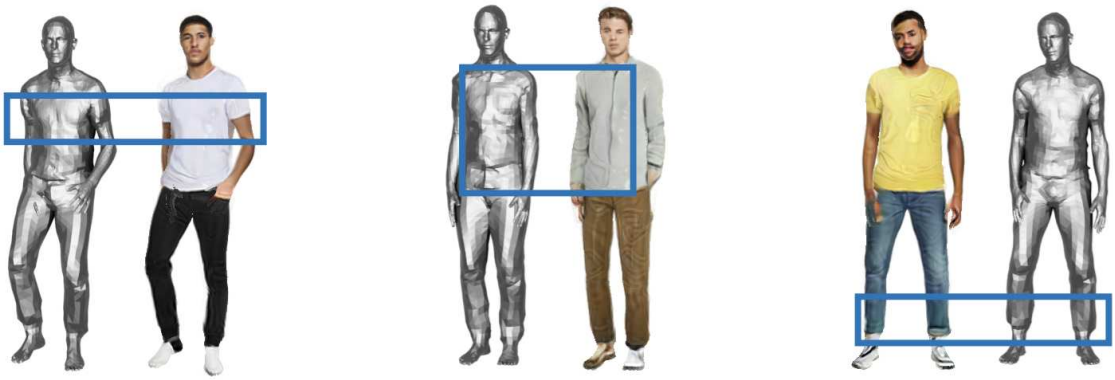


Figure 4.14: **Geometry conforming with texture:** The geometry such as e.g., clothing boundaries, wrinkles, etc. in different body areas (highlighted in blue boxes) are consistent with texture.

Figure 4.15: **Limitations:** The texture quality degrades for out-of-distribution poses (columns 1-3) and for the body's back (right).

4.5 Conclusion

We have introduced SCULPT, a generative model that creates 3D virtual humans using explicit geometry (mesh) and appearance (texture maps). SCULPT represents clothing geometry as offsets from the SMPL body's vertices, and includes a generative model for texture maps based on clothing type and appearance. This approach effectively combines traditional graphics elements such as meshes, forward rendering and texture maps with modern 3D-aware generative models. Our texture model is trained from unpaired 2D-3D data, making it easy to use and retrain on new data. The clothing geometry is learned using a dataset of 3D meshes.The trained model offers generative control over clothing geometry and appearance, with semantic control over clothing type and color, while still retain-

ing SMPL's pose articulation and body-shape variation. Compared to previous 3D-aware generative models, SCULPT offers greater control and produces higher quality geometry and textures.

Chapter 5

Concluding Remarks and Future Directions

5.1 Conclusion

This thesis focuses on using unpaired learning to reconstruct, transpose, and generate digital human avatars, both in 2D and 3D. We do this by utilizing easily obtained labels and making observations based on them. As a result, we develop a set of models that can control various aspects of human geometry, pose, and appearance.

Take RingNet for instance, a model rooted in the concept that individuals maintain consistent facial shape, which varies from person to person. By harnessing the identity labels from 2D facial datasets, RingNet's architecture is capable of estimating a 3D, articulated, and deformable shape from a single 2D image, even without paired 3D training data. It is a flexible approach that distinguishes face pose and expression from shape. Furthermore, we have compiled a new public dataset of accurate 3D head shapes and high-quality images in various conditions, paving the way for enhanced models and novel research directions.

Next is SPICE, designed around the understanding that a person maintains the same 3D body shape across various poses and that 3D body parts that remain visible before and after repositioning should display similar appearance features. To put this understanding into practice, we leverage inexpensive and readily available 3D body models and 3D body shape and pose regressors. As a result, SPICE can generate videos and repose clothed humans from a single image.

Lastly, SCULPT stands on the observation that learning a geometry model typically demands relatively less data than learning an appearance model, as the latter can be quite diverse due to color and pattern variations. This recognition is instrumental in the SCULPT learning process, where the geometry is learned from 3D registrations while appearance is learned from 2D fashion images. Attribute labels obtained from visual question-answering (VQA) models aid the efficient unpaired learning of both geometry and texture. This model creates 3D virtual humans

with explicit control over clothing geometry and appearance. The flexibility of SCULPT's outputs ensures its integration with existing 3D graphics and game engines.

Figure 5.1: **Image generated by DALL-E 2:** Text prompt to generate this image is, "An Indian princess wearing complex, traditional attire stands amidst a futuristic cityscape from a different camera angle. Her clothing is a detailed blend of ancient heritage and futuristic design, featuring intricate embroidery and jewelry. The background showcases a high-tech city with skyscrapers, flying vehicles, and neon lights from a new perspective, emphasizing the princess's connection to both her culture and the advanced world around her. The lighting accentuates her regal stance and the unique fusion of eras in her appearance."

In summary, this thesis improves how we create and control digital humans, making new tools for areas like virtual reality, video games, etc.

Figure 5.2: **Generated by GEN-2 of Runwayml:** Starting from the first frame (top left) and moving along the temporal sequence, the inconsistency in appearance between the first and subsequent frames becomes increasingly apparent. In the final frames, the generated human is anatomically incorrect, as evident in the fingers, hands, and face region.

5.2 Future directions

Although we have made some progress in developing foundation models that can generate images, videos, and text, there are still big challenges when it comes to creating realistic digital humans. These models, especially video foundation models, do well with objects and scenes, but they struggle to make convincing humans. We notice more mistakes with humans because we are used to seeing people and can easily spot anything that looks off, like weird anatomical details or changes in how a person looks.

For instance, when comparing images and videos made by the latest methods, like DALL-E 2 for images (Figure 5.1) and GEN-2 (Esser et al. [2023]) for videos (Figure 5.2), there is a clear difference. DALL-E 2 can make a very detailed and accurate image of a person. But when we try to make a video with GEN-2 using that image and a text description, the video often has problems with making the person's appearance look consistent and realistic.

Right now, video foundation models try to learn from a diffusion process by denoising a bunch of patches of images or video clips (Kondratyuk et al. [2023], Brooks et al. [2024]). However, it is not clear if this is the best way to understand and recreate the complex nature of human bodies. This thesis suggests that maybe we can incorporate information from 3D models of human bodies to help make more accurate and consistent videos of people, i.e. consistent appearance of people. Future research focusing on the integration of foundation models with detailed information from 3D human models, both conceptually and architecturally, will be an interesting direction.

Moving aside from the appearance, representing the 4D geometry of clothed humans is a also significant challenge in computer graphics and animation, encompassing the intricacies of dynamic motion and complex surface details. Traditional methods, such as meshes combined with physics and/or neural (Grigorev et al. [2023]) simulations, offer a way to model these dynamics by separately creating assets like hair, body models, and clothing, and then applying physics to simulate their interactions. While promising, this approach can demand manual engineering and is sensitive to initial conditions, potentially leading to simulation failures.

Implicit surfaces have been identified as proficient in capturing high-detail representations of rigid objects. Considering each frame of a moving human as a rigid object allows implicit functions to preserve intricate details across the body, including clothing, hair, and accessories. However, integrating control over pose into implicit surfaces introduces complexity, especially for articulated models like clothed humans (Chen et al. [2021]). Existing methods often resort to transformations into a canonical pose, directly or indirectly, with complex formulations, yet they may compromise on generalization or detail fidelity during articulation (Chen

et al. [2021], Mihajlovic et al. [2021], Chen et al. [2022]).

Observing the strengths of neural implicit surfaces in representing detailed, frame-wise rigid objects raises the question of a simpler, more effective approach.

Hypernetworks (Ha et al. [2016]), which involve using one neural network to generate the weights for another, present a promising direction. By training neural implicit surfaces for each pose and frame of a particular 4D-clothed human—effectively overfitting to each scenario—these functions can serve as the target networks. A hypernetwork could then be trained, through diffusion or adversarial methods, to produce the weights for these target networks. Additionally, incorporating a control mechanism, such as a "controlnet" (Zhang et al. [2023a]), on top of the hypernetwork could allow for pose-adjusted target network generation.

This approach could lay the groundwork for a foundational model capable of accurately representing 4D clothed human motion, combining the detail preservation of neural implicit surfaces with the adaptability and efficiency of hypernetworks. By simplifying the process and reducing the need for manual engineering, this strategy can hold the potential to advance the modeling of complex, dynamic humans in digital environments.

Appendix A

Additional Qualitative Results of RingNet

Figure A.1: **Reconstruction:** Images are taken from CelebA dataset (Liu et al. [2015]).

Figure A.2: **Reconstruction:** Images are taken from CelebA dataset (Liu et al. [2015]).

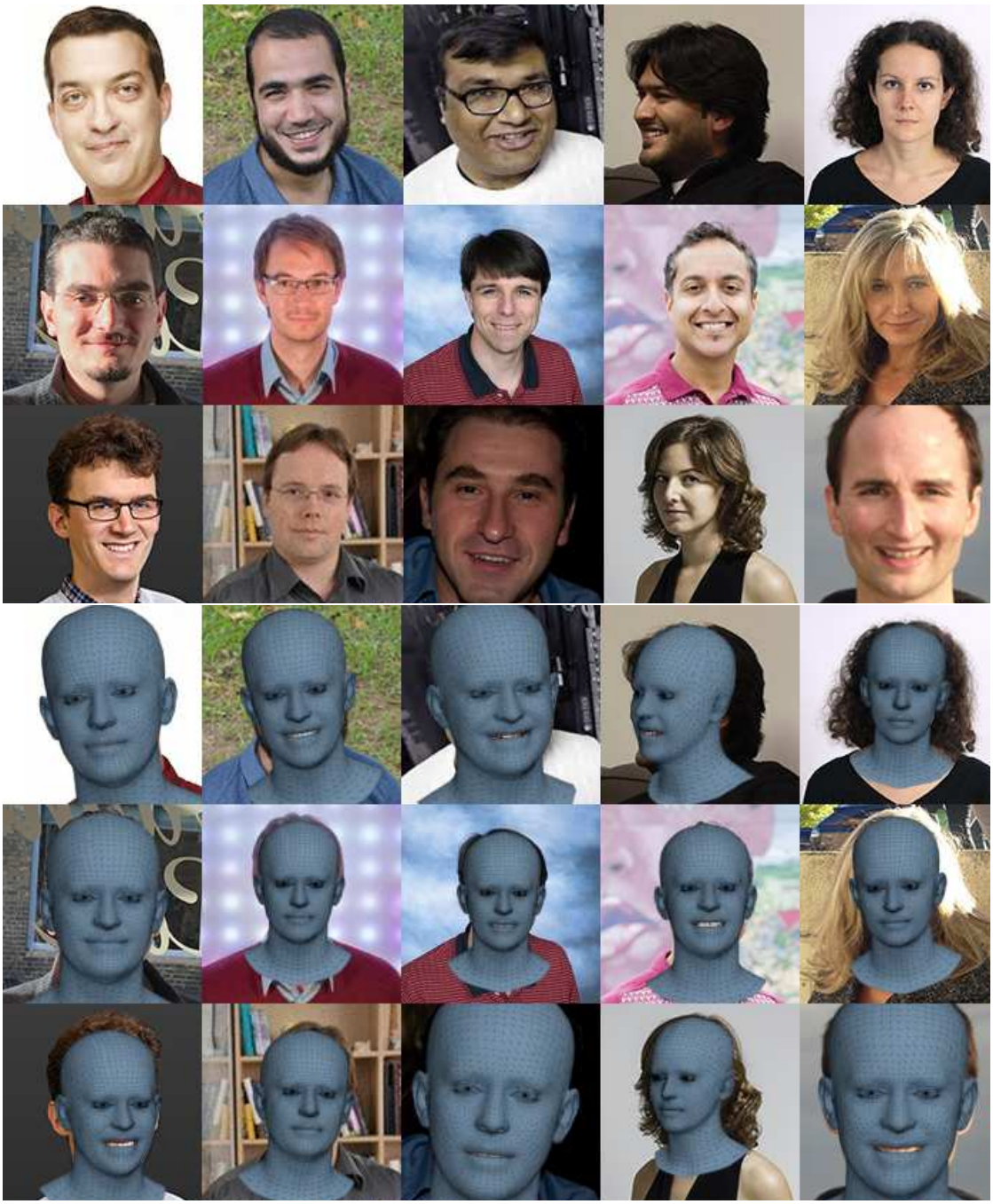


Figure A.3: **Reconstruction:** Images are taken from CVPR 2019 area chairs website.

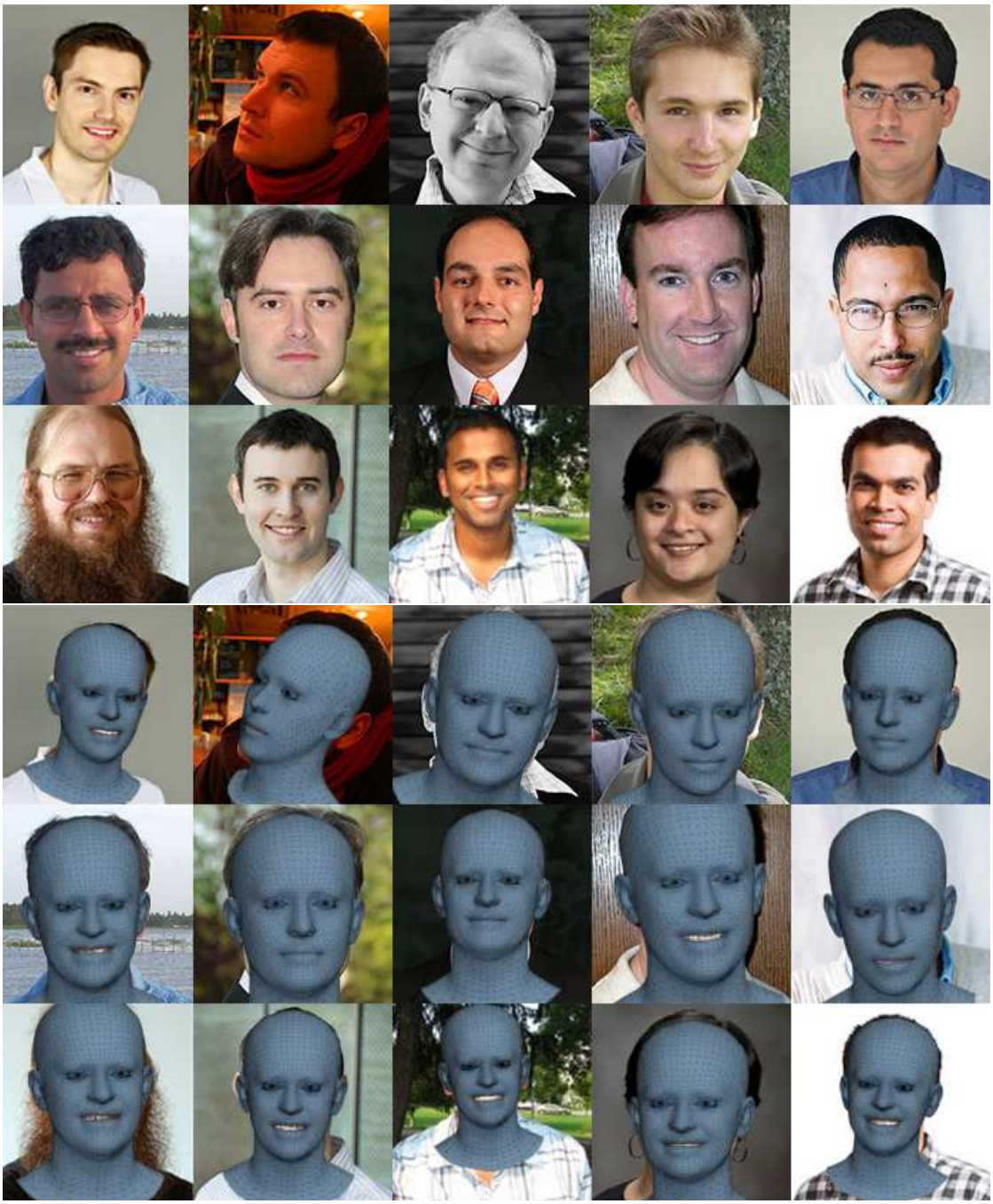


Figure A.4: **Reconstruction:** Images are taken from CVPR 2019 area chairs website.

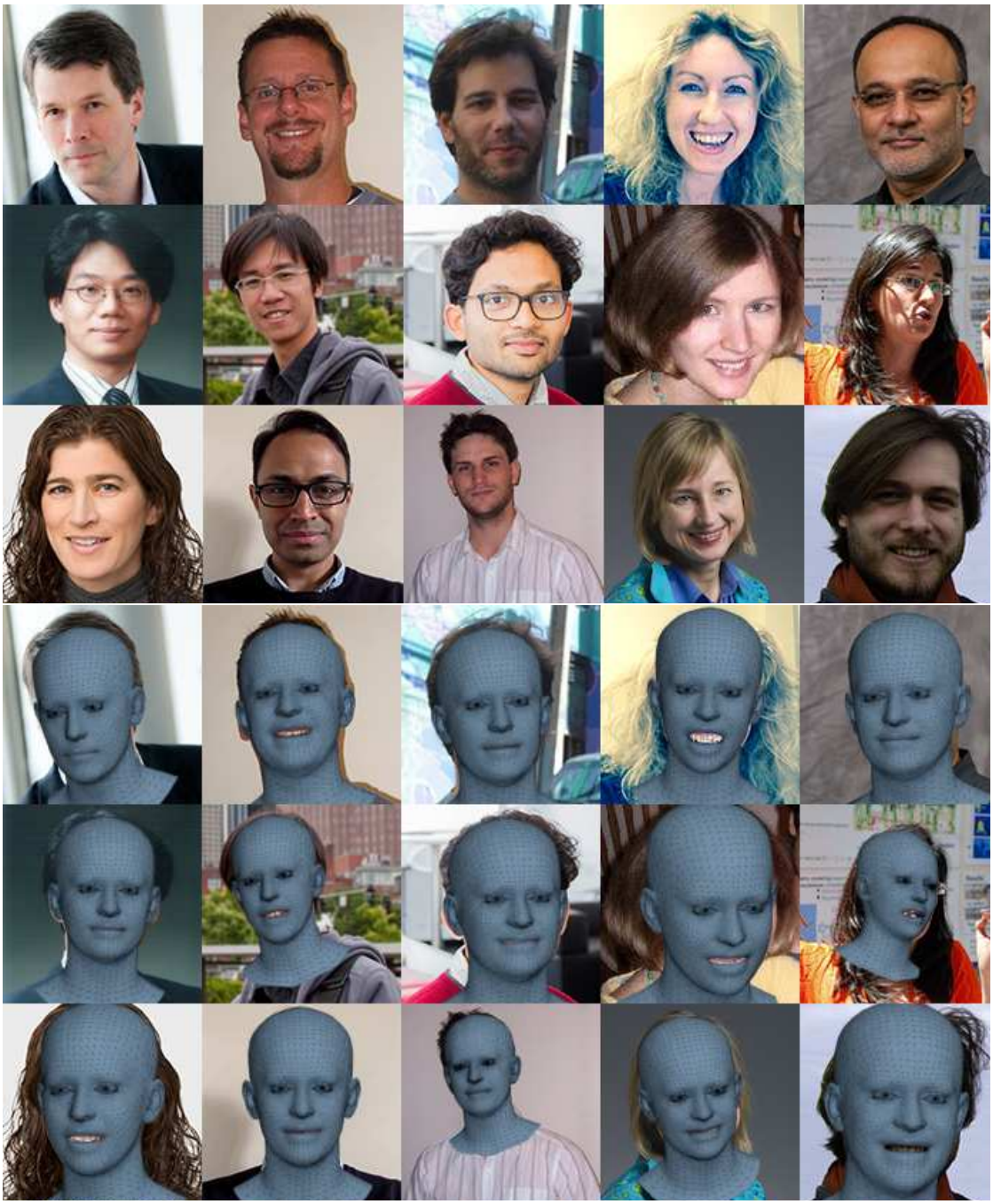


Figure A.5: **Reconstruction:** Images are taken from CVPR 2019 area chairs website.

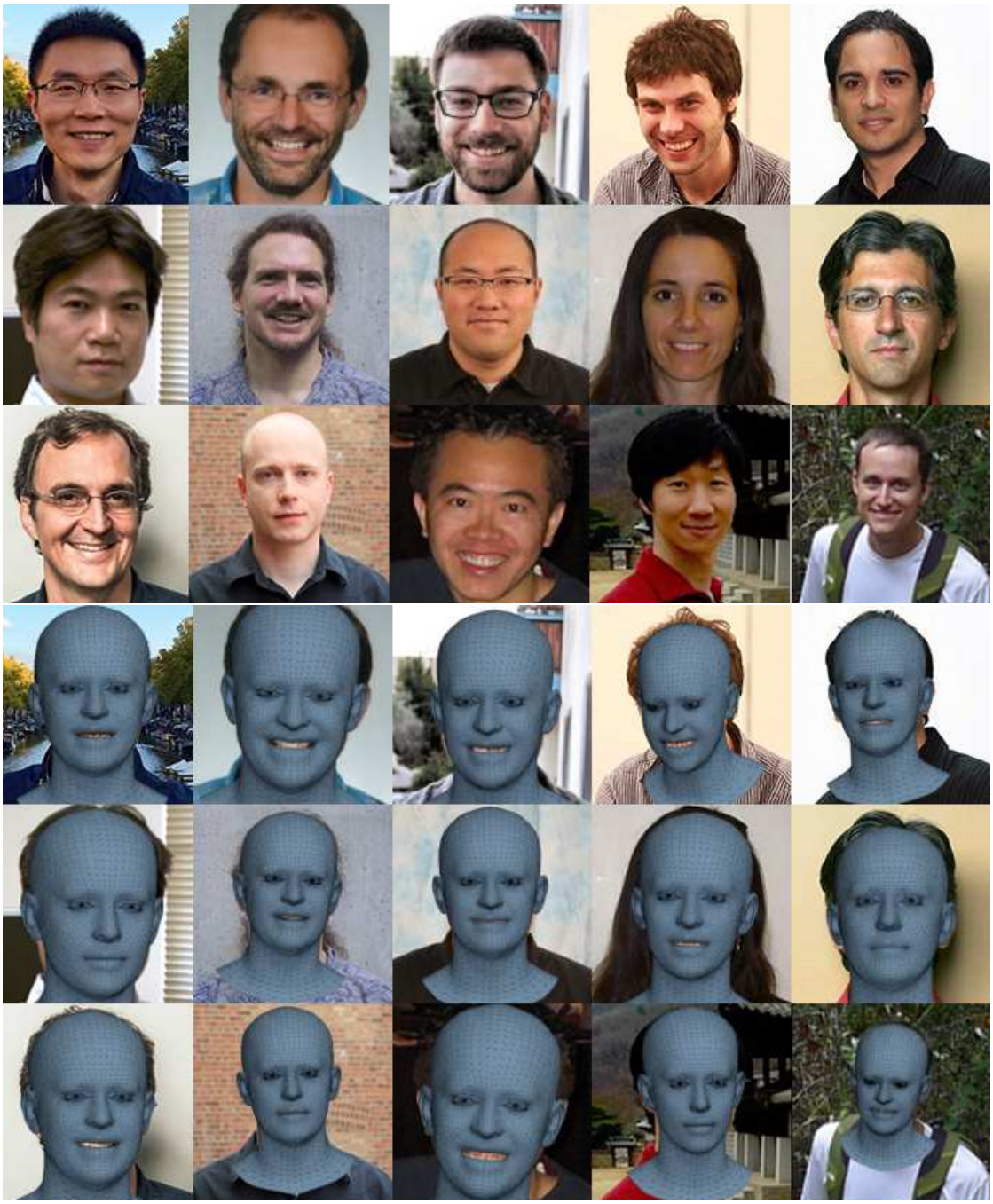


Figure A.6: **Reconstruction:** Images are taken from CVPR 2019 area chairs website.

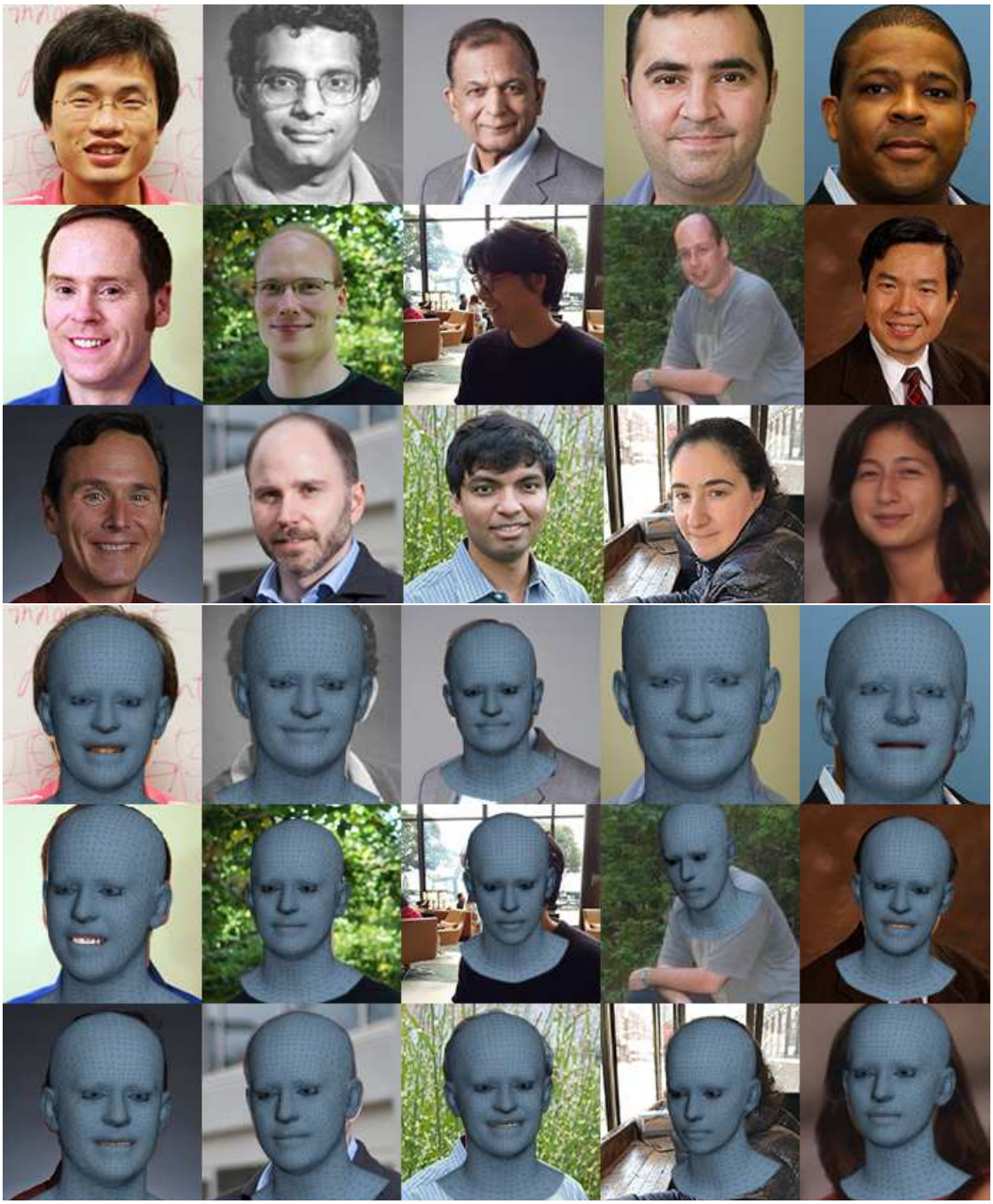


Figure A.7: **Reconstruction:** Images are taken from CVPR 2019 area chairs website.

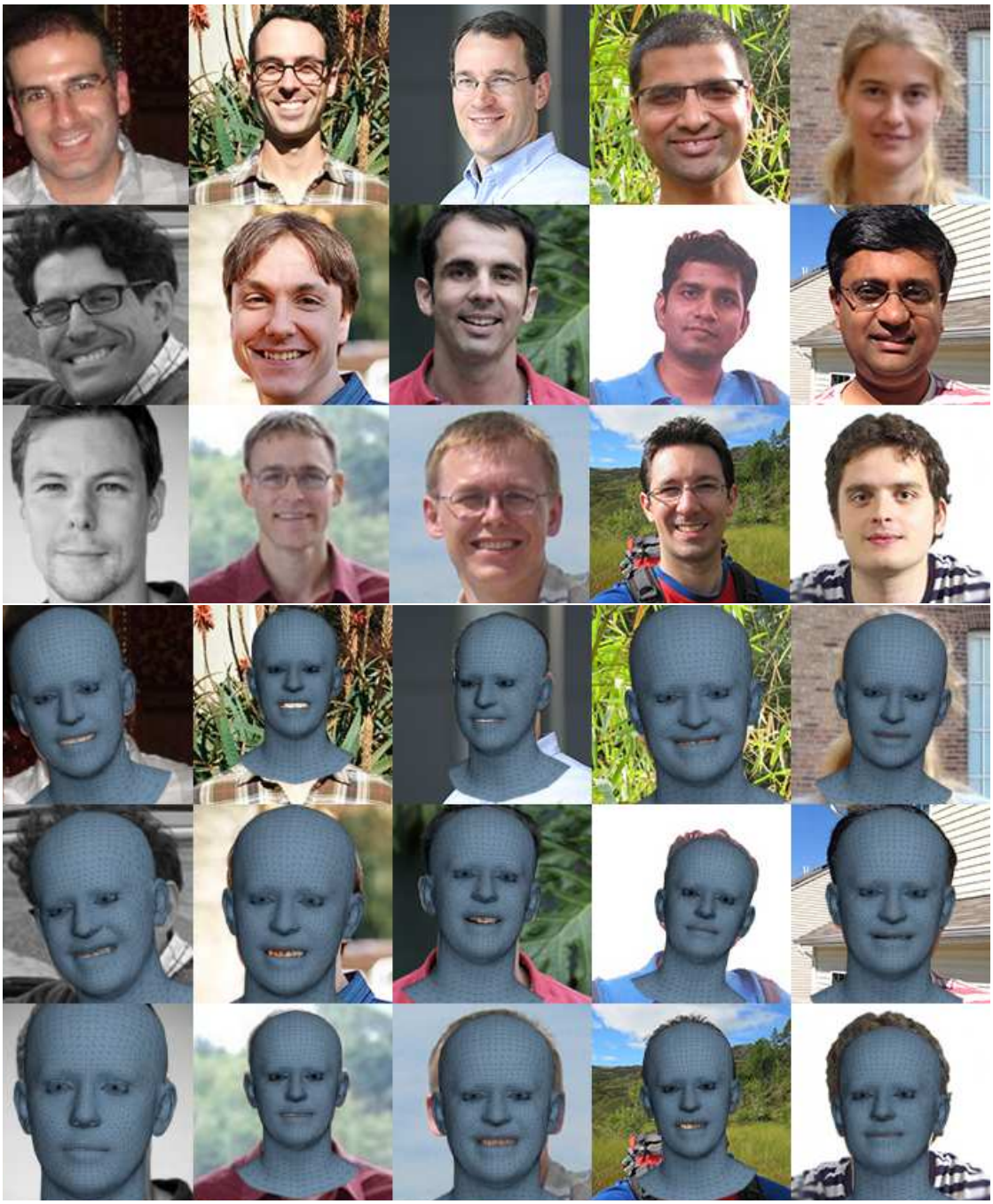


Figure A.8: **Reconstruction:** Images are taken from CVPR 2019 area chairs website.

Appendix B

Sampling correctness and regularization loss for training SPICE

The flow Field Estimator module $\mathcal{G}_F$ of the generator $\mathcal{G}$ of SPICE is trained to predict the flow between P_s and P_t. It takes R_s, P_s, P_t as inputs to generate flow fields f and occlusion masks o_m.

$$f, o_m = \mathcal{G}_F(R_s, P_s, P_t) \tag{B.1}$$

where f denotes the coordinate offsets linking sources to targets, while the occlusion mask o_m, ranging continuously from 0 to 1, signifies the presence of target position information within the sources.

Now the sampling correctness loss calculates the similarity between the warped source feature and ground-truth target feature at the VGG feature level (Simonyan and Zisserman [2015]). Let v_s and v_t represent the feature vectors generated by a specified layer within the VGG19 architecture. The term $v_{s,f} = f(v_s)$ denotes the warped version of the source feature vector v_s, achieved through the application of f. The sampling correctness loss is quantified by computing the relative cosine similarity between the warped source feature vector $v_{s,f}$ and the target feature vector v_t.

$$\mathcal{L}_c = \frac{1}{N} \sum_{l \in \Omega} exp(-\frac{\mu(\mathbf{v}^l_{s,\mathbf{w}}, \mathbf{v}^l_t)}{\mu^l_{max}}) \tag{B.2}$$

where $\mu(*)$ denotes the cosine similarity. Coordinate set Ω contains all N positions in the feature maps, and $\mathbf{v}^l_{s,f}$ denotes the feature of $\mathbf{v}_{s,f}$ located at the coordinate $l = (x, y)$. The normalization term μ^l_{max} is calculated as

$$\mu^l_{max} = \max_{l' \in \Omega} \mu(\mathbf{v}^{l'}_s, \mathbf{v}^l_t) \tag{B.3}$$

Additionally, a regularization term is added to punish local regions where the transformation is not an affine transformation. Let $\mathbf{c}_t$ be the 2D coordinate matrix of the target feature map. The corresponding source coordinate matrix can be written as $\mathbf{c}_s = \mathbf{c}_t + f$. $\mathcal{N}_n(\mathbf{c}_t, l)$ is used to denote local $n \times n$ patch of $\mathbf{c}_t$ centered at the location l. The regularization assumes that the transformation between $\mathcal{N}_n(\mathbf{c}_t, l)$ and $\mathcal{N}_n(\mathbf{c}_s, l)$ is an affine transformation.

$$\mathbf{T}_l = \mathbf{A}_l \mathbf{S}_l = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \mathbf{S}_l \tag{B.4}$$

where $\mathbf{T}_l = \begin{bmatrix} x_1 & x_2 & ... & x_{n\times n} \\ y_1 & y_2 & ... & y_{n\times n} \end{bmatrix}$ with each coordinate $(x_i, y_i) \in \mathcal{N}_n(\mathbf{c}_t, l)$ and $\mathbf{S}_l = \begin{bmatrix} x_1 & x_2 & ... & x_{n\times n} \\ y_1 & y_2 & ... & y_{n\times n} \\ 1 & 1 & ... & 1 \end{bmatrix}$ with each coordinate $(x_i, y_i) \in \mathcal{N}_n(\mathbf{c}_s, l)$. The estimated affine transformation parameters $\hat{\mathbf{A}}_l$ can be solved using the least-squares estimation as

$$\hat{\mathbf{A}}_l = (\mathbf{S}_l^H \mathbf{S}_l)^{-1} \mathbf{S}_l^H \mathbf{T}_l \tag{B.5}$$

and the regularization is calculated as the ℓ_2 distance of the error.

$$\mathcal{L}_r = \sum_{l \in \Omega} \left\| \mathbf{T}_l - \hat{\mathbf{A}}_l \mathbf{S}_l \right\|_2^2 \tag{B.6}$$

Appendix C

Additional Qualitative Results of SPICE

Figure C.1: Additional qualitative results of SPICE. Each triplet in the figure consists of the source image (left), a reference image in target pose (middle) and the generated image in the target pose (right); input and reference images are from the DeepFashion test set (Liu et al. [2016]).

Figure C.2: Additional qualitative results of SPICE. Each triplet in the figure consists of the source image (left), a reference image in target pose (middle) and the generated image in the target pose (right); input and reference images are from the DeepFashion test set (Liu et al. [2016]).

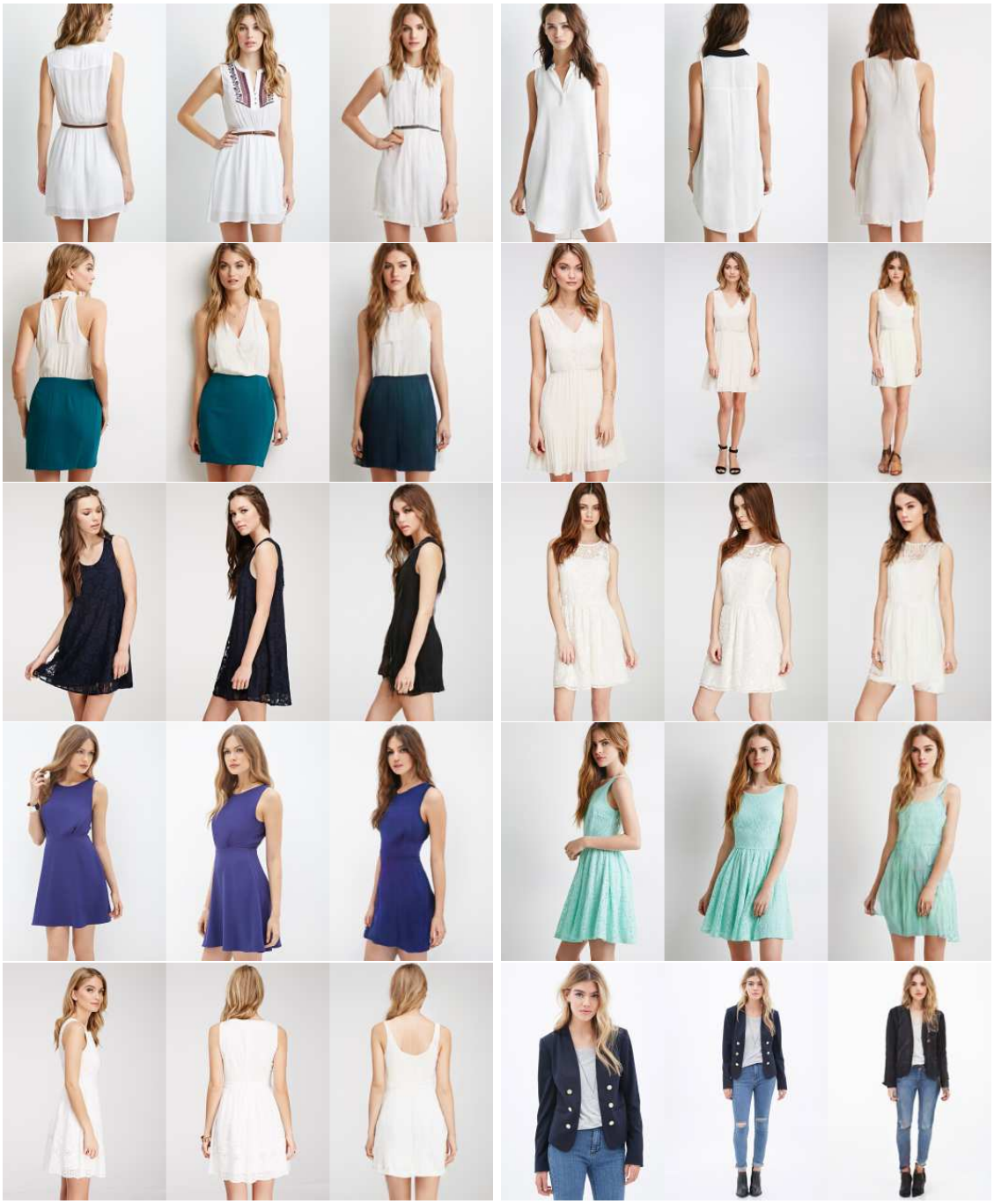


Figure C.3: Additional qualitative results of SPICE. Each triplet in the figure consists of the source image (left), a reference image in target pose (middle) and the generated image in the target pose (right); input and reference images are from the DeepFashion test set (Liu et al. [2016]).

Figure C.4: **Qualitative comparison:** Here we provide additional comparisons of SPICE with other unsupervised learning methods. SPICE outperforms these methods by producing images of superior realism and quality. Additionally, it maintains pose and appearance.

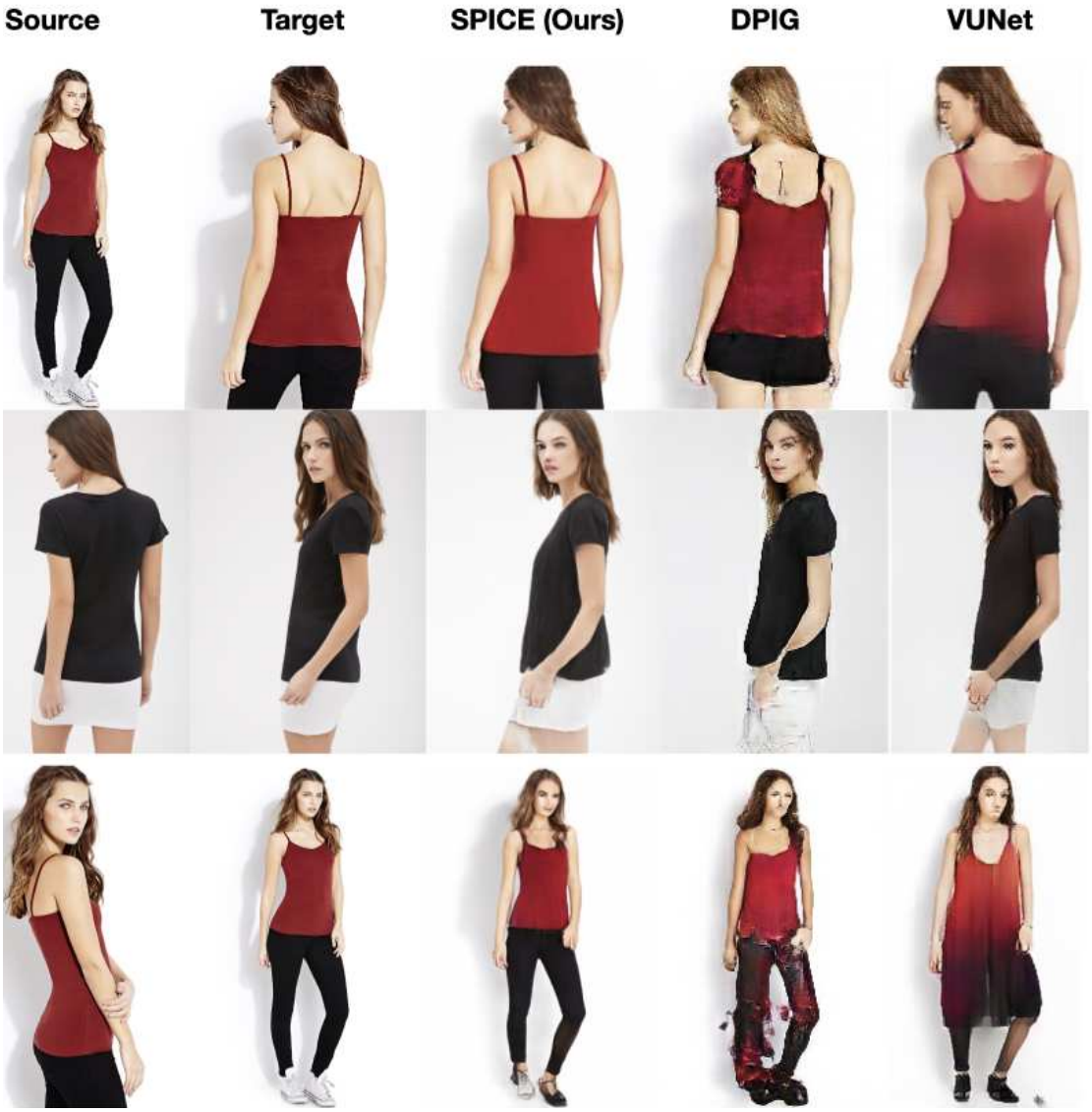


Figure C.5: **Qualitative comparison:** Here we provide additional comparisons of SPICE with other unsupervised learning methods. SPICE outperforms these methods by producing images of superior realism and quality. Additionally, it maintains pose and appearance.

Figure C.6: **Qualitative comparison:** Here we compare SPICE with other supervised learning methods namely Def-GAN (Siarohin et al. [2018a]), Pose-Attn (Zhu et al. [2019]), Intr-Flow (Li et al. [2019]), Ren et al. (Ren et al. [2020]). SPICE performs at par with those.

Appendix D

Additional Qualitative Results of SCULPT

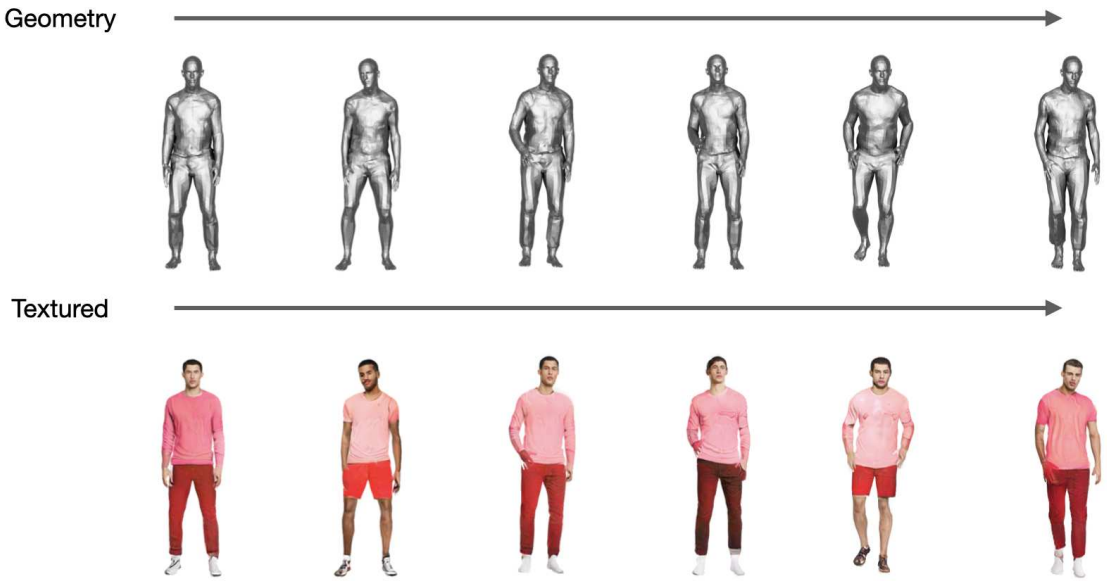


Figure D.1: **Clothing color variation:** $\mathbf{c}_t$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_t$ is CLIP embedding corresponding to the textual input "The color of the upper body clothing is pink and the color of the pants is red".

Figure D.2: **Clothing color variation:** $\mathbf{c}_t$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_t$ is CLIP embedding corresponding to the textual input "The color of the upper body clothing is grey and the color of the pants is black".

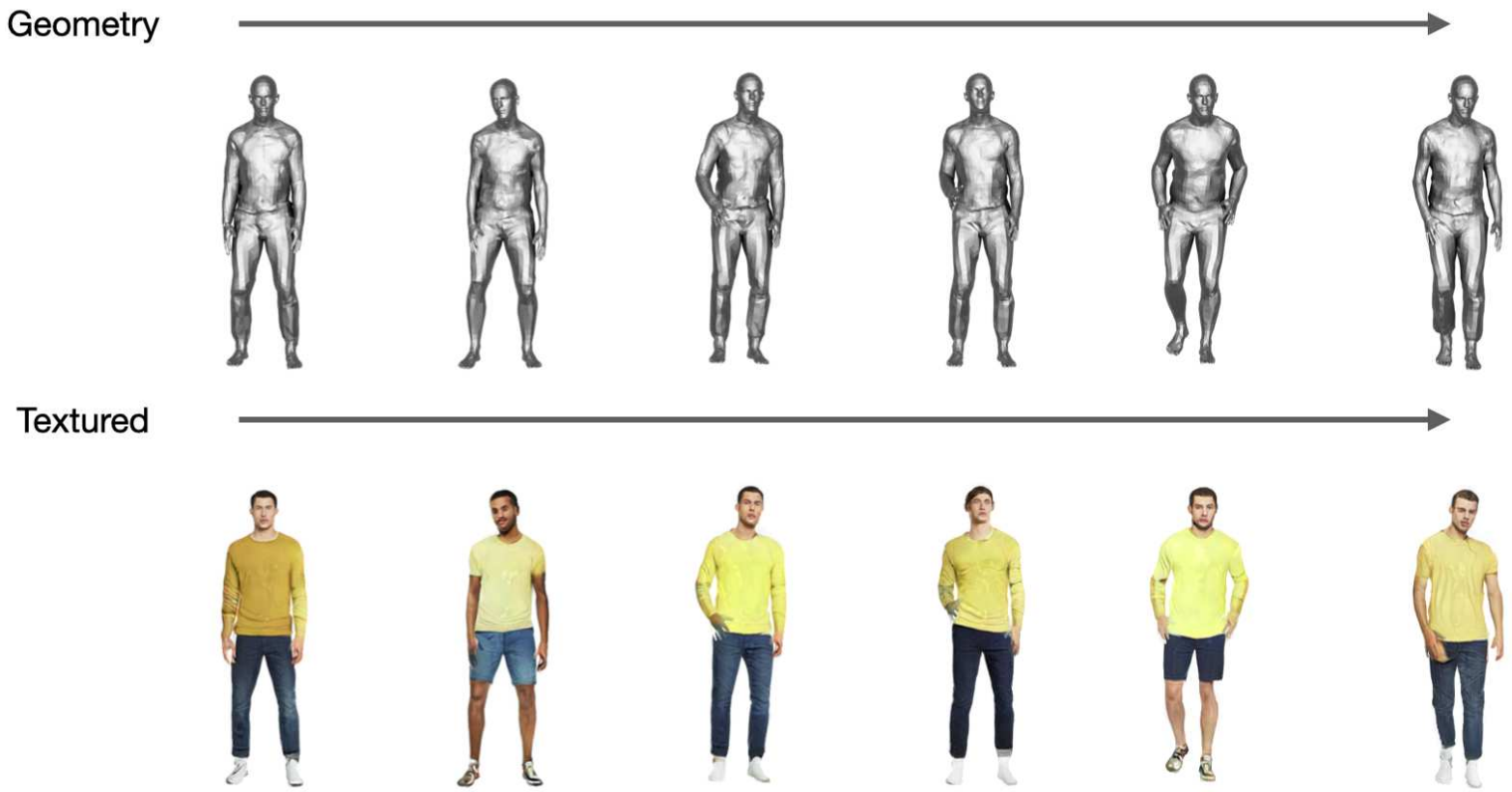


Figure D.3: **Clothing color variation:** $\mathbf{c}_t$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_t$ is CLIP embedding corresponding to the textual input "The color of the upper body clothing is yellow and the color of the pants is blue".

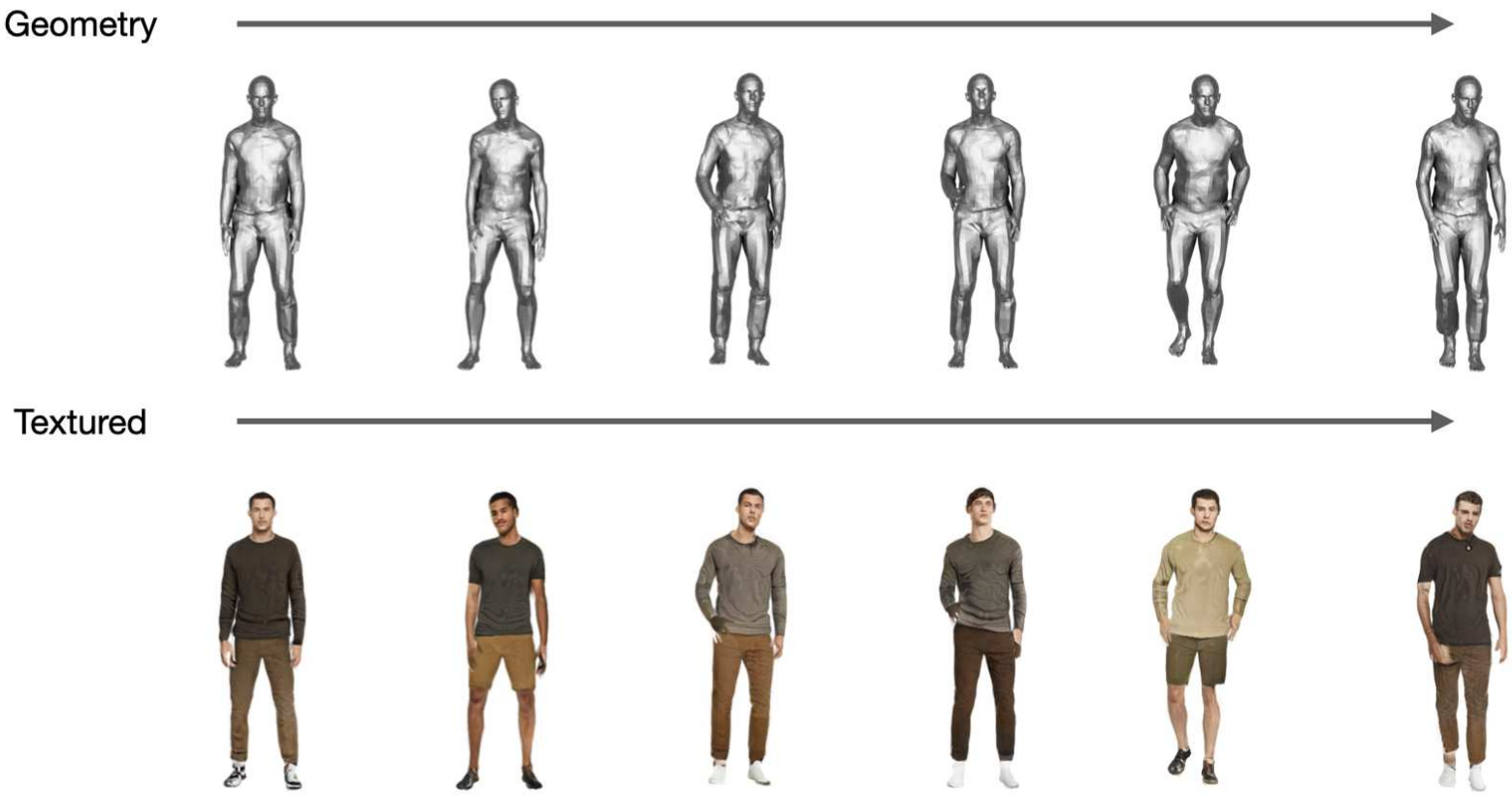


Figure D.4: **Clothing color variation:** $\mathbf{c}_t$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_t$ is CLIP embedding corresponding to the textual input "The color of the upper body clothing is grey and the color of the pants is brown".

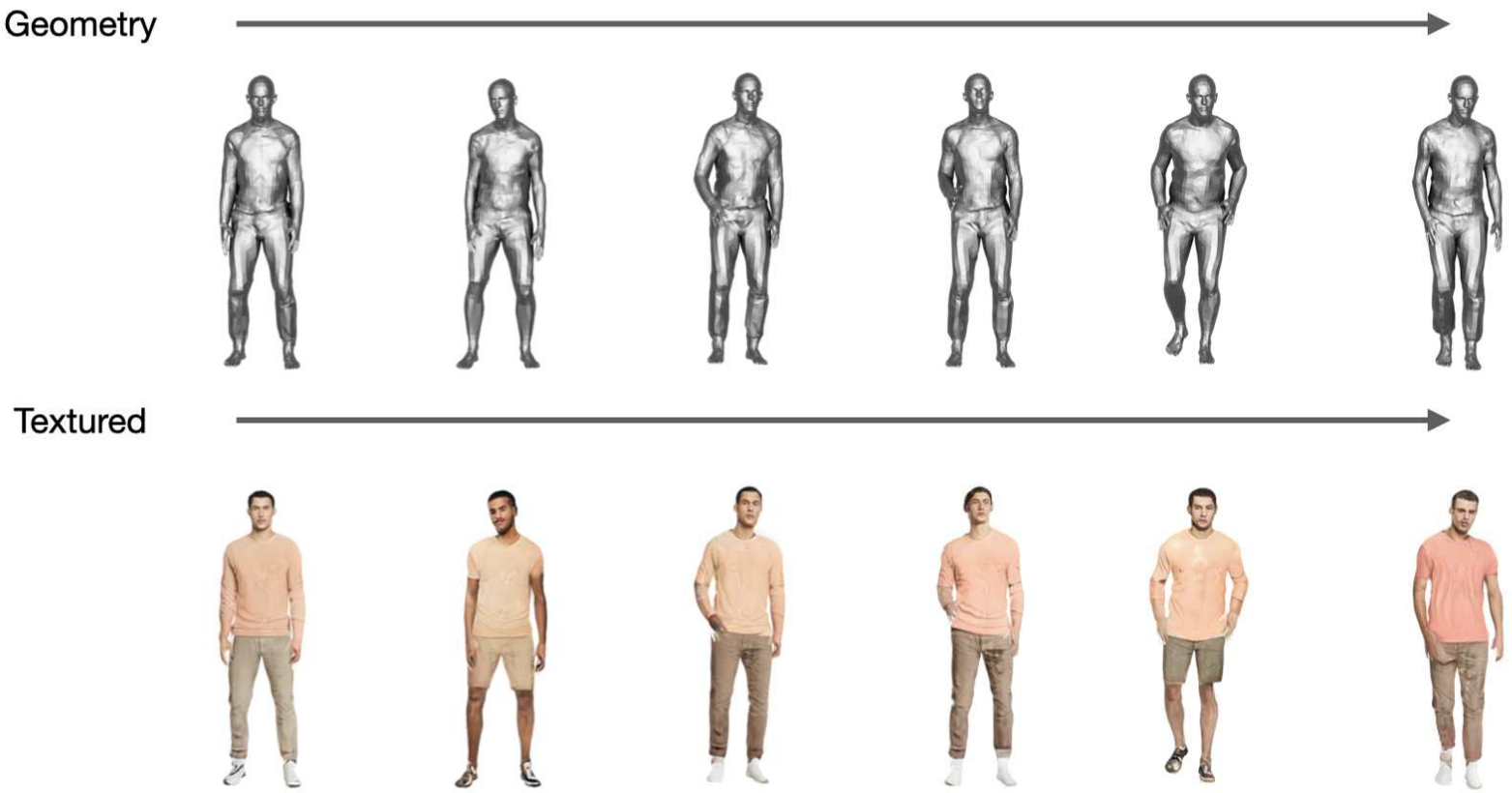


Figure D.5: **Clothing color variation:** $\mathbf{c}_t$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_t$ is CLIP embedding corresponding to the textual input "The color of the upper body clothing is pink and the color of the pants is khaki".

Figure D.6: **Clothing type variation:** $\mathbf{c}_g$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_g$ is the categorical condition vector representing "short sleeve t-shirt/short pants".

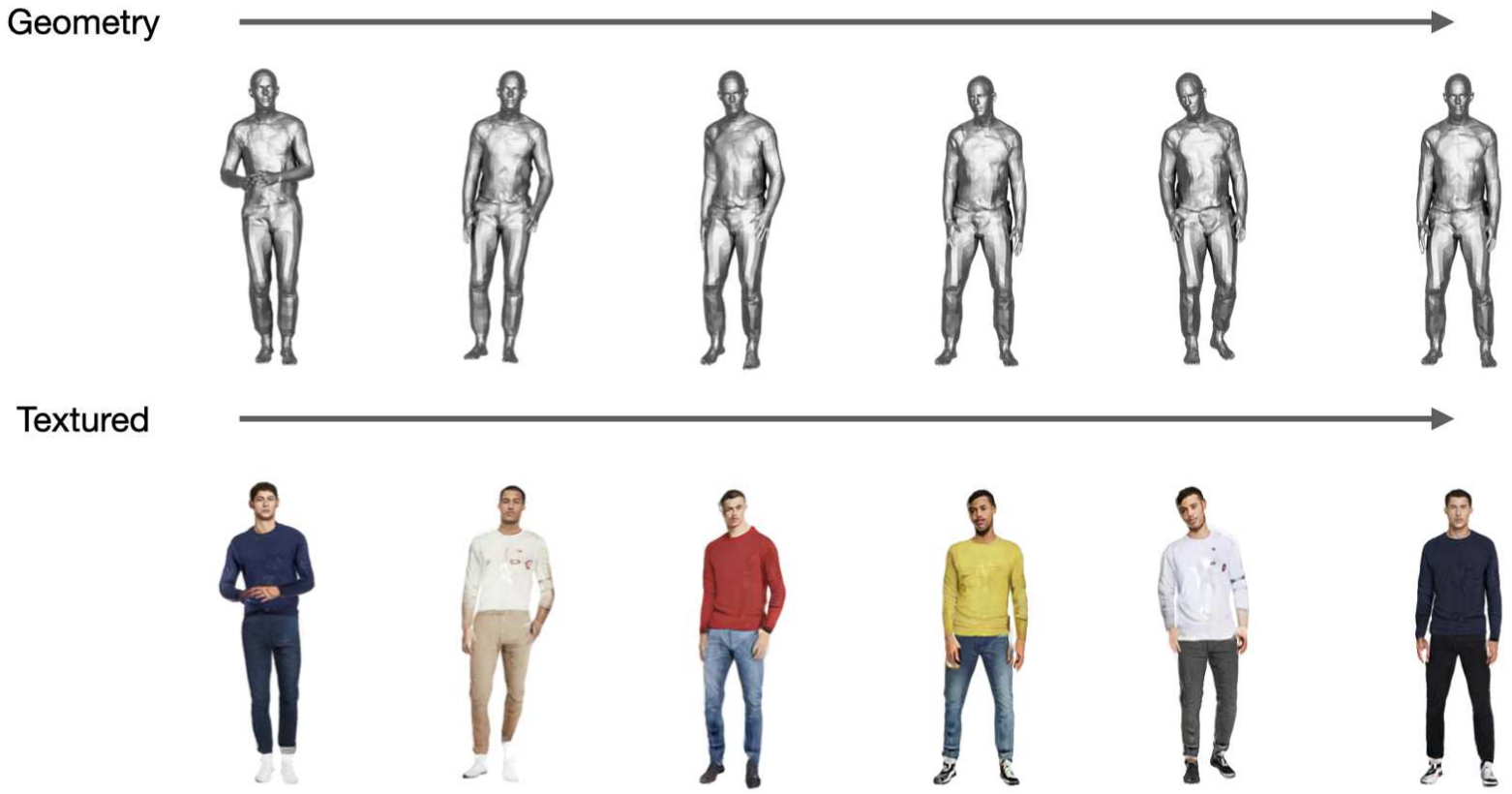


Figure D.7: **Clothing type variation:** $\mathbf{c}_g$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_g$ is the categorical condition vector representing "long sleeve t-shirt/long pants".

Figure D.8: **Clothing type variation: $\mathbf{c}_g$** is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_g$ is the categorical condition vector representing "short sleeve t-shirt/long pants".

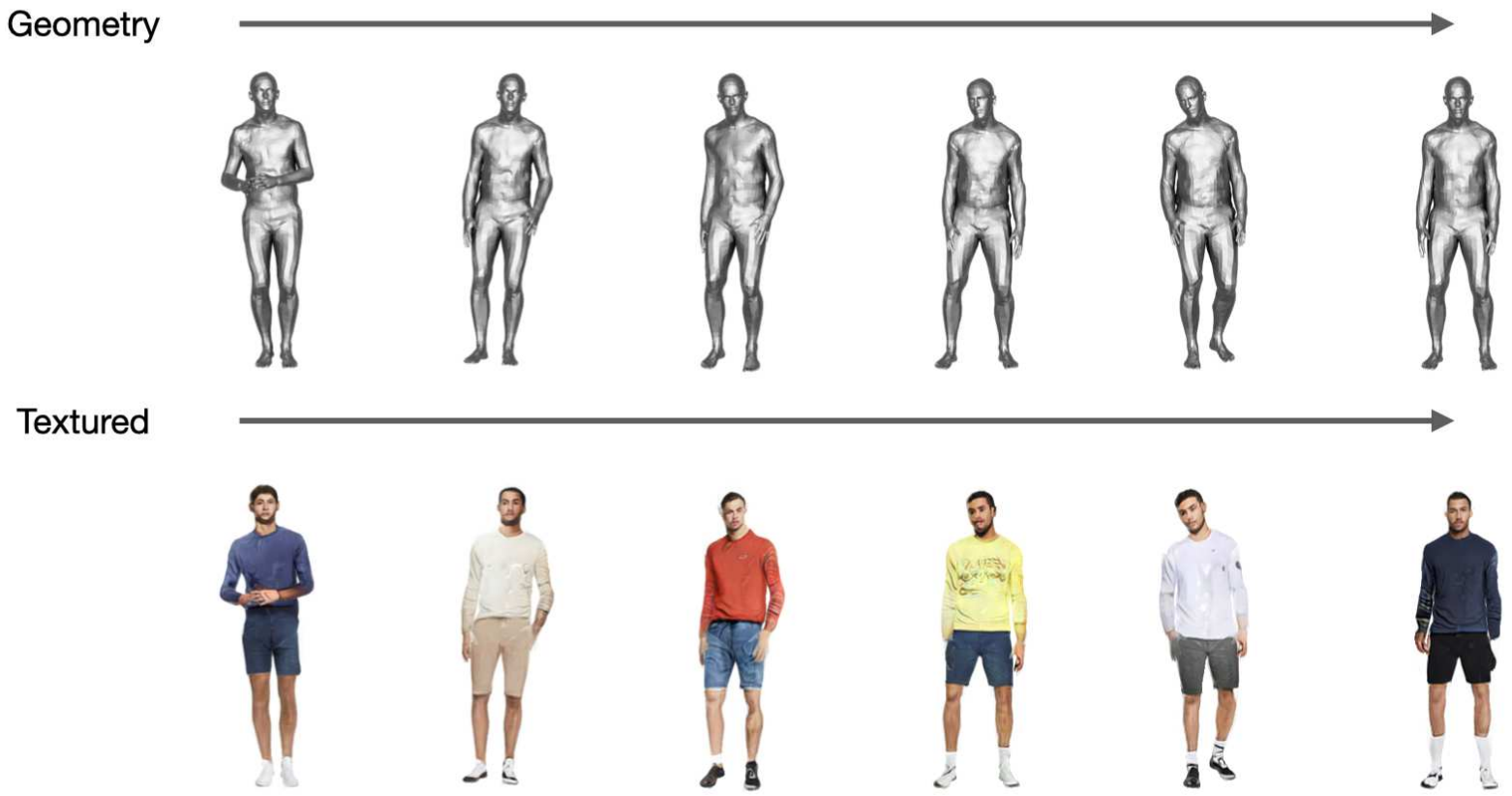


Figure D.9: **Clothing type variation:** $\mathbf{c}_g$ is varied keeping the other factors fixed. The top row shows the geometry whereas the bottom row shows the corresponding textured mesh. For this figure, $\mathbf{c}_g$ is the categorical condition vector representing "long sleeve t-shirt/short pants".

Figure D.10: **Additional qualitative comparisons:** Here, we present additional qualitative comparisons with EG3D (Chan et al. [2022]) (left four columns) and EVA3D (Hong et al. [2023]) (middle four columns). Our method (right four columns) surpasses the performance of these state-of-the-art (SOTA) methods, as demonstrated. Each method is shown with the 3D geometry and the corresponding textured mesh from different viewpoints.

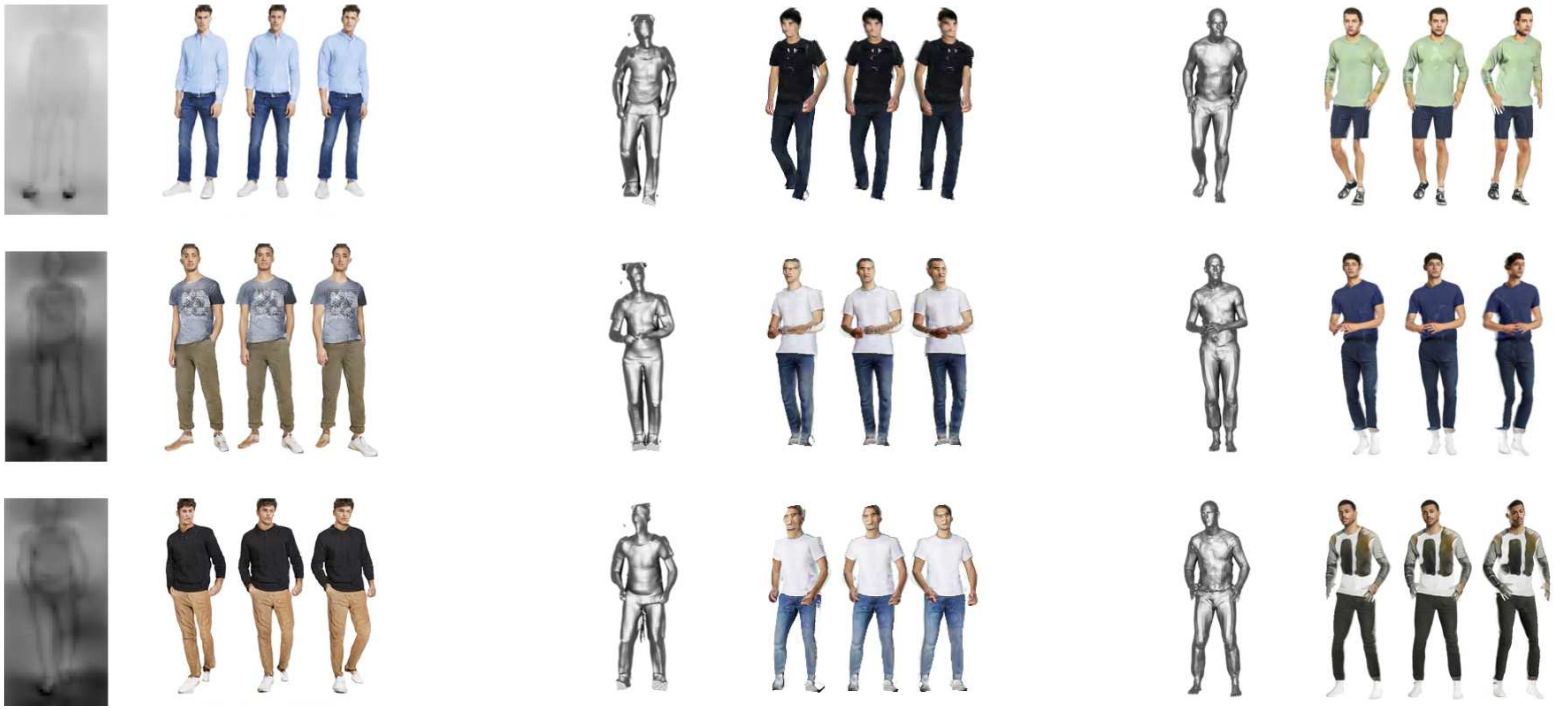


Figure D.11: **Additional qualitative comparisons:** Here, we present additional qualitative comparisons with EG3D (Chan et al. [2022]) (left four columns) and EVA3D (Hong et al. [2023]) (middle four columns). Our method (right four columns) surpasses the performance of these state-of-the-art (SOTA) methods, as demonstrated. Each method is shown with the 3D geometry and the corresponding textured mesh from different viewpoints.

Bibliography

Object keypoints similarity. https://cocodataset.org/#keypoints-eval.

Zalando. https://www.zalando.de, 2021.

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

Badour AlBahar, Jingwan Lu, Jimei Yang, Zhixin Shu, Eli Shechtman, and Jia-Bin Huang. Pose with style: detail-preserving pose-guided image synthesis with conditional StyleGAN. 40(6):218:1–218:11, 2021.

Guha Balakrishnan, Amy Zhao, Adrian V Dalca, Fredo Durand, and John Guttag. Synthesizing images of humans in unseen poses. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8340–8348, 2018.

Anil Bas, William AP Smith, Timo Bolkart, and Stefanie Wuhrer. Fitting a 3D morphable model to edges: A comparison between hard and soft correspondences. In *Asian conference on Computer Vision (ECCV)*, pages 377–391, 2016.

Alexander W. Bergman, Petr Kellnhofer, Yifan Wang, Eric R. Chan, David B. Lindell, and Gordon Wetzstein. Generative neural articulated radiance fields. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Soma Biswas, Kevin W Bowyer, and Patrick J Flynn. Multidimensional scaling for matching low-resolution face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34(10):2019–2030, 2011.

Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. *In Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.

Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D

human pose and shape from a single image. In *European Conference on Computer Vision (ECCV)*, Lecture Notes in Computer Science. Springer International Publishing, October 2016.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. URL `https://openai.com/research/video-generation-models-as-world-simulators`.

Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial landmarks). In *International Conference on Computer Vision (ICCV)*, 2017.

Chen Cao, Qiming Hou, and Kun Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Trans. Graph.*, 33(4):43:1–43:10, July 2014. ISSN 0730-0301. doi: 10.1145/2601097.2601204. URL `http://doi.acm.org/10.1145/2601097.2601204`.

Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *International proceedings on Automatic Face & Gesture Recognition (FG)*, 2018.

Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2019.

Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4733–4742, 2016.

Eric R. Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. PI-GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5799–5809, 2021.

Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16102–16112, 2022.

Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit

shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11594–11604, 2021.

Xu Chen, Tianjian Jiang, Jie Song, Jinlong Yang, Michael J. Black, Andreas Geiger, and Otmar Hilliges. gDNA: Towards generative detailed neural avatars. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20427–20437, 2022.

Enric Corona, Albert Pumarola, Guillem Alenya, Gerard Pons-Moll, and Francesc Moreno-Noguer. Smplicit: Topology-aware generative model for clothed people. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11875–11885, 2021.

Hang Dai, Nick Pears, William A. P. Smith, and Christian Duncan. A 3d morphable model of craniofacial shape and texture variation. In *International Conference on Computer Vision (ICCV)*, Oct 2017.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4690–4699, 2019a.

Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019b.

Haoye Dong, Xiaodan Liang, Ke Gong, Hanjiang Lai, Jia Zhu, and Jian Yin. Soft-gated warping-GAN for pose-guided person image synthesis. *Advances in Neural Information Processing Systems (NeurIPS)*, 31:472–482, 2018.

Haoye Dong, Xiaodan Liang, Yixuan Zhang, Xujie Zhang, Xiaohui Shen, Zhenyu Xie, Bowen Wu, and Jian Yin. Fashion editing with adversarial parsing learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020a.

Haoye Dong, Xiaodan Liang, Yixuan Zhang, Xujie Zhang, Xiaohui Shen, Zhenyu Xie, Bowen Wu, and Jian Yin. Fashion editing with adversarial parsing learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8120–8128, 2020b.

Patrick Esser, Ekaterina Sutter, and Björn Ommer. A variational U-Net for conditional appearance and shape generation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8857–8866, 2018.

Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023.

Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3D face reconstruction and dense alignment with position map regression network. In *European Conference on Computer Vision (ECCV)*, 2018a.

Yao Feng, Haiwen Feng, Michael J Black, and Timo Bolkart. Learning an animatable detailed 3d face model from in-the-wild images. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 40(4):1–13, 2021.

Zhen-Hua Feng, Patrik Huber, Josef Kittler, Peter Hancock, Xiao-Jun Wu, Qijun Zhao, Paul Koppen, and Matthias Rätsch. Evaluation of dense 3D reconstruction from 2D face images in the wild. In *International proceedings on Automatic Face & Gesture Recognition (FG)*, pages 780–786, 2018b.

Jianglin Fu, Shikai Li, Yuming Jiang, Kwan-Yee Lin, Chen Qian, Chen Change Loy, Wayne Wu, and Ziwei Liu. StyleGAN-Human: A data-centric Odyssey of human generation. In *European Conference on Computer Vision (ECCV)*, volume 13676, pages 1–19, 2022.

Thittaporn Ganokratanaa, Supavadee Aramvith, and Nicu Sebe. Unsupervised anomaly detection and localization based on deep spatiotemporal translation network. *IEEE Access*, 8:50312–50329, 03 2020. doi: 10.1109/ACCESS.2020.2979869.

Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. GET3D: A generative model of high quality 3D textured shapes learned from images. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Pablo Garrido, Michael Zollhöfer, Dan Casas, Levi Valgaerts, Kiran Varanasi, Patrick Pérez, and Christian Theobalt. Reconstruction of personalized 3D face rigs from monocular video. *ACM Transactions on Graphics (TOG)*, 35(3), 2016.

Kyle Genova, Forrester Cole, Aaron Maschinot, Aaron Sarna, Daniel Vlasic, and William T Freeman. Unsupervised training for 3D morphable model regressiond. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8377–8386, 2018.

Thomas Gerig, Andreas Morel-Forster, Clemens Blumer, Bernhard Egger, Marcel Luthi, Sandro Schönborn, and Thomas Vetter. Morphable face models-an

open framework. In *International proceedings on Automatic Face & Gesture Recognition (FG)*, pages 75–82, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2672–2680, 2014.

Artur Grigorev, Artem Sevastopolsky, Alexander Vakhitov, and Victor Lempitsky. Coordinate-based texture inpainting for pose-guided human image generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12135–12144, 2019.

Artur Grigorev, Karim Iskakov, Anastasia Ianina, Renat Bashirov, Ilya Zakharkin, Alexander Vakhitov, and Victor Lempitsky. StylePeople: A generative model of fullbody human avatars. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5151–5160, 2021.

Artur Grigorev, Michael J Black, and Otmar Hilliges. Hood: Hierarchical graphs for generalized modelling of clothing dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16965–16974, 2023.

Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010.

Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7297–7306, 2018.

David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision (ECCV)*, pages 630–645, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6626–6637, 2017.

Fangzhou Hong, Zhaoxi Chen, Yushi LAN, Liang Pan, and Ziwei Liu. EVA3D: Compositional 3D human generation from 2D image collections. In *International Conference on Learning Representations (ICLR)*, 2023. URL `https://openreview.net/forum?id=g7U9jD_2CUr`.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Aaron S Jackson, Adrian Bulat, Vasileios Argyriou, and Georgios Tzimiropoulos. Large pose 3D face reconstruction from a single image via direct volumetric cnn regression. In *International Conference on Computer Vision (ICCV)*, 2017.

Suyi Jiang, Haoran Jiang, Ziyu Wang, Haimin Luo, Wenzheng Chen, and Lan Xu. HumanGen: Generating human radiance fields with explicit priors. *arXiv preprint arXiv:2212.05321*, 2022.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, pages 694–711. Springer, 2016.

Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4401–4410, 2019.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:12104–12114, 2020a.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8110–8119, 2020b.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 852–863, 2021.

Zhanghan Ke, Jiayu Sun, Kaican Li, Qiong Yan, and Rynson W.H. Lau. MODNet: Real-time trimap-free portrait matting via objective decomposition. In

Proceedings of the AAAI proceedings on Artificial Intelligence, pages 1140–1147, 2022.

Ira Kemelmacher-Shlizerman and Steven M Seitz. Face reconstruction in the wild. In *International Conference on Computer Vision (ICCV)*, pages 1746–1753, 2011.

Hyeongwoo Kim, Michael Zollhöfer, Ayush Tewari, Justus Thies, Christian Richardt, and Christian Theobalt. Inversefacenet: Deep monocular inverse face rendering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4625–46342, 2018.

Minchul Kim, Anil K Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18750–18759, 2022.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015a.

Diederik P Kingma and Jimmy Ba. A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, volume 5, 2015b.

Markus Knoche, István Sárándi, and Bastian Leibe. Reposing humans by warping 3D features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1044–1045, 2020.

Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *International Conference on Computer Vision (ICCV)*, October 2019a.

Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *International Conference on Computer Vision (ICCV)*, 2019b.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Rachel Hornung, Hartwig Adam, Hassan Akbari, Yair Alon, Vighnesh Birodkar, et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023.

Mohamed Ilyes Lakhal, Oswald Lanz, and Andrea Cavallaro. Pose guided human image synthesis by view disentanglement and enhanced weighting loss. In *European Conference on Computer Vision (ECCV)*, pages 380–394. Springer, 2018.

Chunlu Li, Andreas Morel-Forster, Thomas Vetter, Bernhard Egger, and Adam Kortylewski. Robust model-based face reconstruction through weakly-supervised outlier segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 372–381, 2023.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International proceedings on Machine Learning (ICML)*, volume 162, pages 12888–12900, 2022.

Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics (TOG)*, 36(6):194, 2017a.

Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017b. URL `https://doi.org/10.1145/3130800.3130813`.

Yining Li, Chen Huang, and Chen Change Loy. Dense intrinsic appearance flow for human pose transfer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3693–3702, 2019.

Wen Liu, Zhixin Piao, Jie Min, Wenhan Luo, Lin Ma, and Shenghua Gao. Liquid warping gan: A unified framework for human motion imitation, appearance transfer and novel view synthesis. In *International Conference on Computer Vision (ICCV)*, pages 5904–5913, 2019.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaogang Tang. Deep learning face attributes in the wild. In *International Conference on Computer Vision (ICCV)*, December 2015.

Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 406–416, 2017.

Liqian Ma, Qianru Sun, Stamatios Georgoulis, Luc Van Gool, Bernt Schiele, and Mario Fritz. Disentangled person image generation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 99–108, 2018.

Liqian Ma, Zhe Lin, Connelly Barnes, Alexei A Efros, and Jingwan Lu. Unselfie: Translating selfies to neutral-pose portraits in the wild. In *European Conference on Computer Vision (ECCV)*, pages 156–173. Springer, 2020a.

Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learning to dress 3D people in generative clothing. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6468–6477, 2020b.

Yifang Men. CX score source code. `https://github.com/menyifang/ADGAN`.

Yifang Men, Yiming Mao, Yuning Jiang, Wei-Ying Ma, and Zhouhui Lian. Controllable person image synthesis with attribute-decomposed gan. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5084–5093, 2020.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for GANs do actually converge? In *International proceedings on Machine Learning (ICML)*, volume 80, pages 3481–3490, 2018.

Marko Mihajlovic, Yan Zhang, Michael J Black, and Siyu Tang. Leap: Learning articulated occupancy of people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10461–10471, 2021.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, pages 405–421. Springer, 2020.

Sivaram Prasad Mudunuri and Soma Biswas. Low resolution face recognition across variations in pose and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 38(5):1034–1040, 2015.

Natalia Neverova, Riza Alp Guler, and Iasonas Kokkinos. Dense pose transfer. In *European Conference on Computer Vision (ECCV)*, pages 123–138, 2018.

Atsuhiro Noguchi, Xiao Sun, Stephen Lin, and Tatsuya Harada. Unsupervised learning of efficient geometry-aware neural articulated representations. In *European Conference on Computer Vision (ECCV)*, pages 597–614. Springer, 2022.

Roy Or-El, Xuan Luo, Mengyi Shan, Eli Shechtman, Jeong Joon Park, and Ira Kemelmacher-Shlizerman. StyleSDF: High-resolution 3D-consistent image and geometry generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13503–13513, 2022.

Pablo Palafox, Aljaž Božič, Justus Thies, Matthias Nießner, and Angela Dai. NPMs: Neural parametric models for 3D deformable shapes. In *International Conference on Computer Vision (ICCV)*, pages 12695–12705, 2021.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 8024–8035, 2019.

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019.

Albert Pumarola, Antonio Agudo, Alberto Sanfeliu, and Francesc Moreno-Noguer. Unsupervised person image synthesis in arbitrary poses. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8620–8628, 2018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International proceedings on Machine Learning (ICML)*, pages 8748–8763, 2021.

Aashish Rai, Hiresh Gupta, Ayush Pandey, Francisco Vicente Carrasco, Shingo Jason Takagi, Amaury Aubel, Daeil Kim, Aayush Prakash, and Fernando De la Torre. Towards realistic generative 3d face models. In *Winter proceedings on Applications of Computer Vision (WACV)*, pages 3738–3748, 2024.

Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. Generating 3D faces using convolutional mesh autoencoders. In *European Conference on Computer Vision (ECCV)*, volume Lecture Notes in Computer Science, vol 11207, pages 725–741. Springer, Cham, September 2018. URL `http://coma.is.tue.mpg.de/`.

Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3D deep learning with pytorch3D. *arXiv:2007.08501*, 2020.

Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H. Li, and Ge Li. Deep image spatial transformation for person image generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Mehdi SM Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

Soubhik Sanyal, Sivaram Prasad Mudunuri, and Soma Biswas. Discriminative pose-free descriptors for face and object matching. In *International Conference on Computer Vision (ICCV)*, pages 3837–3845, 2015.

Soubhik Sanyal, Devraj Mandal, and Soma Biswas. Aligned discriminative pose robust descriptors for face and object recognition. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 820–824. IEEE, 2017a.

Soubhik Sanyal, Sivaram Prasad Mudunuri, and Soma Biswas. Discriminative pose-free descriptors for face and object matching. *Pattern Recognition*, 67: 353–365, 2017b.

Soubhik Sanyal, Timo Bolkart, Haiwen Feng, and Michael Black. Learning to regress 3D face shape and expression from an image without 3D supervision. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7763–7772, June 2019.

Soubhik Sanyal, Alex Vorobiov, Timo Bolkart, Matthew Loper, Betty Mohler, Larry S. Davis, Javier Romero, and Michael J. Black. Learning realistic human reposing using cyclic self-supervision with 3D shape, pose, and appearance consistency. In *International Conference on Computer Vision (ICCV)*, pages 11138–11147, 2021.

Soubhik Sanyal, Partha Ghosh, Jinlong Yang, Michael J. Black, Justus Thies, and Timo Bolkart. SCULPT: Shape-conditioned unpaired learning of pose-dependent clothed and textured human meshes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Kripasindhu Sarkar, Dushyant Mehta, Weipeng Xu, Vladislav Golyanik, and Christian Theobalt. Neural re-rendering of humans from a single image. In *European Conference on Computer Vision (ECCV)*, 2020.

Kripasindhu Sarkar, Vladislav Golyanik, Lingjie Liu, and Christian Theobalt. Style and pose control for image synthesis of humans from a single monocular view. *arXiv preprint arXiv:2102.11263*, 2021a.

Kripasindhu Sarkar, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. HumanGAN: A generative model of human images. In *International proceedings on 3D Vision (3DV)*, pages 258–267, 2021b.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015.

Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. GRAF: Generative radiance fields for 3D-aware image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 20154–20166, 2020.

Katja Schwarz, Axel Sauer, Michael Niemeyer, Yiyi Liao, and Andreas Geiger. VoxGRAF: Fast 3D-aware image synthesis with sparse voxel grids. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. `https://github.com/mseitzer/pytorch-fid`, August 2020. Version 0.1.1.

Matan Sela, Elad Richardson, and Ron Kimmel. Unrestricted facial geometry reconstruction using image-to-image translation. In *International Conference on Computer Vision (ICCV)*, pages 1585–1594, 2017.

Soumyadip Sengupta, Angjoo Kanazawa, Carlos D Castillo, and David W Jacobs. Sfsnet: Learning shape, reflectance and illuminance of faces in the wild. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Haoyue Shi, Le Wang, Wei Tang, Nanning Zheng, and Gang Hua. Loss functions for person image generation. In *Proceedings of the British Machine Vision inproceedings (BMVC)*, 2020.

Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuiliere, and Nicu Sebe. Deformable gans for pose-based human image generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3408–3416, 2018a.

Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuilière, and Nicu Sebe. Deformable gans for pose-based human image generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018b.

Aliaksandr Siarohin, Stéphane Lathuilière, Enver Sangineto, and Nicu Sebe. Appearance and pose-conditioned human image generation using deformable gans. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2019.

Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*, 2015.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 7462–7473, 2020.

Sijie Song, Wei Zhang, Jiaying Liu, and Tao Mei. Unsupervised person image generation with semantic parsing transformation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Supasorn Suwajanakorn, Ira Kemelmacher-Shlizerman, and Steven M Seitz. Total moving face reconstruction. In *European Conference on Computer Vision (ECCV)*, pages 796–812, 2014.

Hao Tang, Song Bai, Li Zhang, Philip HS Torr, and Nicu Sebe. Xinggan for person image generation. In *European Conference on Computer Vision (ECCV)*, 2020.

Jilin Tang, Yi Yuan, Tianjia Shao, Yong Liu, Mengmeng Wang, and Kun Zhou. Structure-aware person image generation with pose decomposition and semantic correlation. In *Proceedings of the AAAI proceedings on Artificial Intelligence*, 2021.

Ayush Tewari, Michael Zollhofer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Christian Theobalt. MoFA: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *International Conference on Computer Vision (ICCV)*, pages 1274–1283, 2017.

Ayush Tewari, Michael Zollhöfer, Pablo Garrido, Florian Bernard, Hyeongwoo Kim, Patrick Pérez, and Christian Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2387–2395, 2016.

Anh Tuân Trân, Tal Hassner, Iacopo Masi, Eran Paz, Yuval Nirkin, and Gérard Medioni. Extreme 3D face reconstruction: Seeing through occlusions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3935–3944, 2018.

Luan Tran and Xiaoming Liu. Nonlinear 3D face morphable model. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Anh Tuan Tran, Tal Hassner, Iacopo Masi, and Gérard Medioni. Regressing robust and discriminative 3D morphable models with a very deep neural network. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1493–1502, 2017.

Mei Wang and Weihong Deng. Deep face recognition: A survey. *Neurocomputing*, 429:215–244, 2021.

Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1473–1480, 2006.

Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3681–3691, 2021.

Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Matthew Johnson, Jingjing Shen, Nikola Milosavljević, Daniel Wilde, Stephan Garbin, Toby Sharp, Ivan Stojiljković, et al. 3d face reconstruction with dense landmarks. In *European Conference on Computer Vision (ECCV)*, pages 160–177. Springer, 2022.

Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J. Black. ICON: Implicit clothed humans obtained from normals. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13296–13306, 2022.

Ceyuan Yang, Zhe Wang, Xinge Zhu, Chen Huang, Jianping Shi, and Dahua Lin. Pose guided human video generation. In *European Conference on Computer Vision (ECCV)*, pages 201–216, 2018.

Hongtao Yang, Tong Zhang, Wenbing Huang, Xuming He, and Fatih Porikli. Towards purely unsupervised disentanglement of appearance and shape for person images generation. In *Proceedings of the 1st International Workshop on Human-centric Multimedia Analysis*, pages 33–41, 2020.

Lingbo Yang, Pan Wang, Chang Liu, Zhanning Gao, Peiran Ren, Xinfeng Zhang, Shanshe Wang, Siwei Ma, Xiansheng Hua, and Wen Gao. Towards fine-grained human pose transfer with detail replenishing network. *IEEE Transactions on Image Processing*, 30:2422–2435, 2021.

Wenwu Yang, Yeqing Zhao, Bailin Yang, and Jianbing Shen. Learning 3d face reconstruction from the cycle-consistency of dynamic faces. *IEEE Transactions on Multimedia*, 2023.

Polina Zablotskaia, Aliaksandr Siarohin, Bo Zhao, and Leonid Sigal. Dwnet: Dense warp-based network for pose-guided human video generation. *Proceedings of the British Machine Vision inproceedings (BMVC)*, 2019.

Mihai Zanfir, Elisabeta Oneata, Alin-Ionut Popa, Andrei Zanfir, and Cristian Sminchisescu. Human synthesis and scene compositing. In *Proceedings of the AAAI proceedings on Artificial Intelligence*, volume 34, pages 12749–12756, 2020.

Jianfeng Zhang, Zihang Jiang, Dingdong Yang, Hongyi Xu, Yichun Shi, Guoxian Song, Zhongcong Xu, Xinchao Wang, and Jiashi Feng. AvatarGen: a 3D generative model for animatable human avatars. *arXiv preprint arXiv:2208.00561*, 2022.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *International Conference on Computer Vision (ICCV)*, pages 3836–3847, 2023a.

Pan Zhang, Bo Zhang, Dong Chen, Lu Yuan, and Fang Wen. Cross-domain correspondence learning for exemplar-based image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5143–5153, 2020.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. lpips: LPIPS metric for PyTorch. `https://github.com/richzhang/PerceptualSimilarity`.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z. Li. Sfd: Single shot scale-invariant face detector. 2017.

Tianke Zhang, Xuangeng Chu, Yunfei Liu, Lijian Lin, Zhendong Yang, Zhengzhuo Xu, Chengkun Cao, Fei Yu, Changyin Zhou, Chun Yuan, et al. Accurate 3d face reconstruction with facial component tokens. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9033–9042, 2023b.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5745–5753, 2019.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *International Conference on Computer Vision (ICCV)*, 2017.

Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 146–155, 2016.

Zhen Zhu, Tengteng Huang, Baoguang Shi, Miao Yu, Bofei Wang, and Xiang Bai. Progressive pose attention transfer for person image generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2347–2356, 2019.

Wojciech Zielonka, Timo Bolkart, and Justus Thies. Towards metrical reconstruction of human faces. In *European Conference on Computer Vision (ECCV)*, pages 250–269. Springer, 2022.

Michael Zollhöfer, Justus Thies, Pablo Garrido, Derek Bradley, Thabo Beeler, Patrick Pérez, Marc Stamminger, Matthias Nießner, and Christian Theobalt. State of the art on monocular 3d face reconstruction, tracking, and applications. In *Computer graphics forum*, volume 37, pages 523–550. Wiley Online Library, 2018.